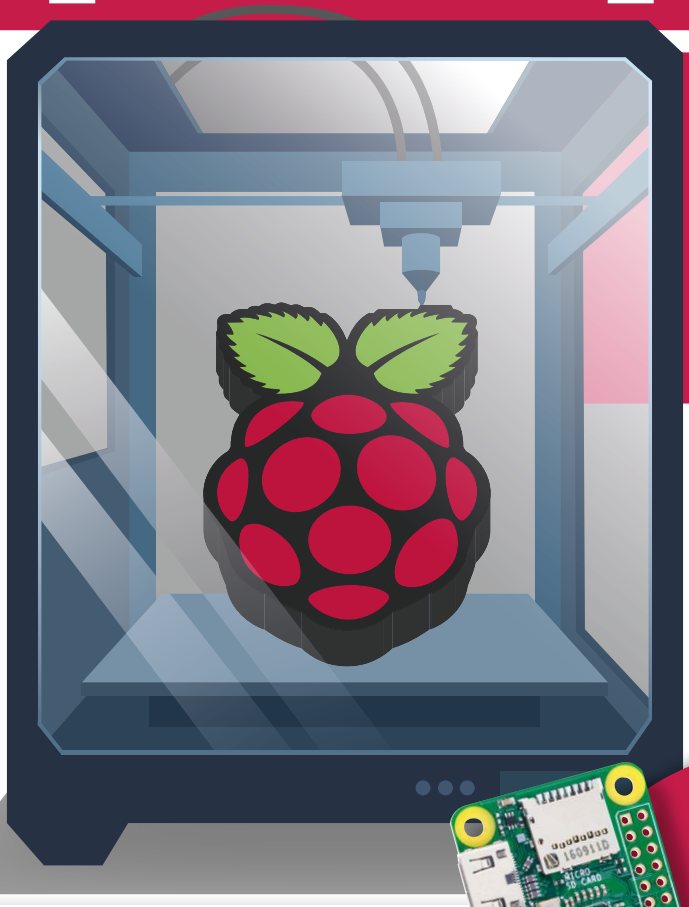
BUY IN PRINT WORLDWIDE! magpi.cc/store

The MagPi

The official Raspberry Pi magazine | Issue 69 | May 2018 | raspberrypi.org/magpi

AFFORDABLE 3D PRINTING

- Buy your first 3D printer
- Use OctoPrint for easy control
- All the greatest 3D projects



STREAM YOUR MUSIC

Set up Bluetooth on your Raspberry Pi

NEW GOOGLE AIY KITS

Now with Pi Zero WH included!

FROM £99

TRANSFORM A RETRO CAM

Kodak Brownie adapted into a modern digicam

PI ZERO TV STICK

Upgrade any TV into a PC with a modified Pi Zero W

Also inside:

- PLAYING TETRIS ON A GIANT LED MATRIX TABLE
- TAKE SCIENTIFIC MEASUREMENTS WITH MAGNETS
- GOLD PANNING EVENT JUDGED WITH RASPBERRY PI
- USING A RASPBERRY PI TO PERFORM FINE EMBROIDERY

GLOBAL DELIVERY
magpi.cc/store
Raspberry Pi PRESS

CanaKit™

CanaKit Raspberry Pi 3 Ultimate Starter Kit

Model B | 1 GB RAM | 1.2 GHz | Quad-Core CPU

> Learn to Code
> Explore Computing
> Get started with Electronics

Kit Includes Raspberry Pi 3 and ...

- Premium Case & Heat Sinks

- 2.5A Power Adapter

- 32 GB Class 10 MicroSD Card

Pre-Loaded with Operating System

- USB MicroSD Card Reader

- Premium HDMI Cable

- Quick-Start Guide

- GPIO to Breadboard Interface Board

- Ribbon Cable

- Full-Size Breadboard

- Jumpers

Male to Male & Male to Female

- LEDs

- Resistors & Push-Buttons

Available for worldwide shipping at:

WWW.CANAKIT.COM

Raspberry Pi Zero W

Now available at CanaKit!

Wi Fi™ Bluetooth®

Electronic Kits • Electronic Parts • Raspberry Pi • Arduino

WELCOME TO THE OFFICIAL MAGAZINE

The last few months have seen a rollercoaster-style drop in the price of 3D printers. These devices, used to create a dizzying array of parts, are still at the cutting edge of maker technology. Owning a 3D printer is simply cool.

Until now, owning a 3D printer has also been very expensive. Which is why they've mostly been found in makerspaces and schools. So, if you've ever wanted a 3D printer, but figured they were too expensive, then now is the time to think again.

Hooking up a low-cost 3D printer to a Raspberry Pi is a great idea. The Raspberry Pi acts as a controller, downloading the 3D printer files and controlling the print process. There's also a huge range of 3D-printed items that have been created especially for the Raspberry Pi. Our 3D printing feature (page 16) has everything you need to get started with 3D printing.

The Raspberry Pi has changed the way we think about computing. And the new 3D printers can change how we think about making. We're really excited by the more democratic nature of 3D printing and hope you find it fun too. Let us know how you get on.

Lucy Hattersley
Editor

YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
AFFORDABLE 3D PRINTING
STREAM YOUR MUSIC
NEW GOOGLE AIY KITS
FROM £99
TRANSFORM A RETRO CAM
PI ZERO TV STICK
GET A PI ZERO W WITH A SUBSCRIPTION! PAGE 26
SUBSCRIBE FROM £4
SEE PAGE 26 FOR DETAILS

THIS MONTH:

16 AFFORDABLE 3D PRINTING
Creating your own printed wonders is cheaper than ever

28 LED MATRIX TABLE
Try eating your dinner off this pixellated marvel

36 SOFI ROBOT FISH
A totally Pi-powered piscine robot

62 PI ZERO TV STICK
Give any television Raspberry Pi smarts

FIND US ONLINE raspberrypi.org/magpi **GET IN TOUCH** magpi@raspberrypi.org

The MagPi

Available on the App Store

GET IT ON Google Play

EDITORIAL
Editor: **Lucy Hattersley**
lucy@raspberrypi.org
Features Editor: **Rob Zwetsloot**
rob.zwetsloot@raspberrypi.org
Sub Editors: **Phil King and Jem Roberts**

DESIGN
Critical Media: **criticalmedia.co.uk**
Head of Design: **Dougal Matthews**
Designers: **Mike Kay and Lee Allen**
Illustrator: **Sam Alder**

PUBLISHING
For advertising & licensing:
Publishing Director: **Russell Barnes**
russell@raspberrypi.org | **+44 (0)7904 766523**
Director of Communications: **Liz Upton**
CEO: **Eben Upton**

DISTRIBUTION
Seymour Distribution Ltd
2 East Poultry Ave
London
EC1A 9PT | **+44 (0)207 429 4000**

SUBSCRIPTIONS
Raspberry Pi Press
Mann Enterprises, Unit E, Brocks Business Centre, Haverhill, CB9 8QP
magpi.cc/subscribe

CONTRIBUTORS
3Dami, b3d101, Alex Bate, Brian Beuken, Mike Cook, David Crookes, Monique Dewanchand, PJ Evans, Tom Haines, Gareth Halfacree, Peter Juett, Peter Kemp, Nicola King, Matt Richardson, Richard Smedley, Clive Webster

CC BY NC SA

recycle
When you have finished with this magazine, please recycle it.

This magazine is printed on paper sourced from sustainable forests and the printer operates an environmental management system which has been assessed as conforming to ISO 14001.

The MagPi magazine is published by Raspberry Pi (Trading) Ltd., 30 Station Road, Cambridge, CB1 2JH. The publisher, editor, and contributors accept no responsibility in respect of any omissions or errors relating to goods, products or services referred to or advertised in the magazine. Except where otherwise noted, content in this magazine is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0). ISSN: 2051-9982.

Raspberry Pi PRESS

Contents

Issue 69 May 2018 raspberrypi.org/magpi

TUTORIALS

> PI 101: BLUETOOTH SPEAKERS **38**
Connect Bluetooth audio devices to your Raspberry Pi

> MAKE A MAGNETIC BOUNCER **40**
Create a weird interface with the Pi Bakery

> CREATE AN INFO SCREEN **46**
A system of for creating house-wide touchscreens

> BUILD A 3D ROCKET MODEL **50**
Learn how to make printable 3D models

> MAKE GAMES IN C PART 5 **54**
Now we can start making an actual game

IN THE NEWS

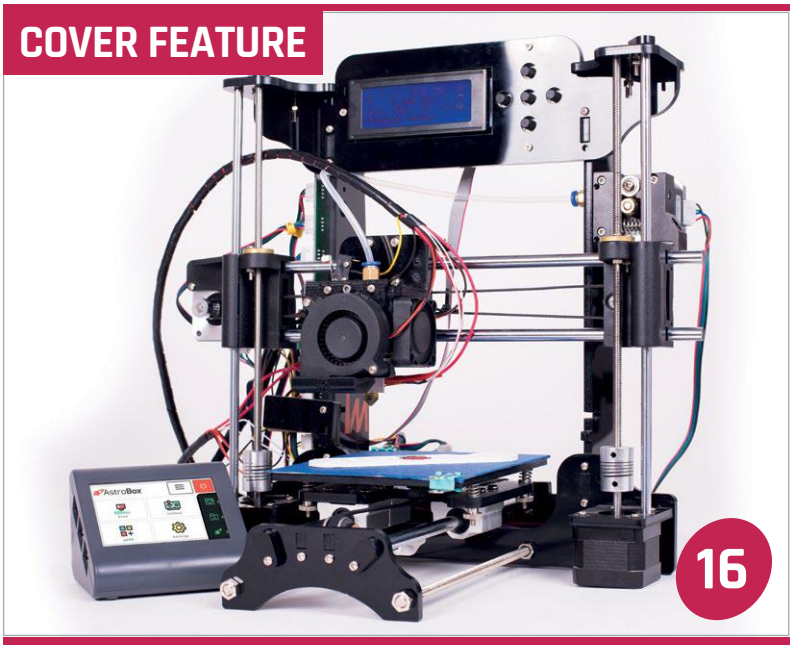
NEW AIY KITS

Upgraded AIY Projects kits out now! **06**

FORMULA PI WINNER

New champions announced! **08**

COVER FEATURE



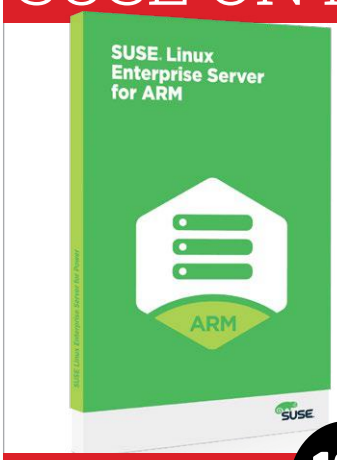
16

AFFORDABLE 3D PRINTING

ASTRO PI

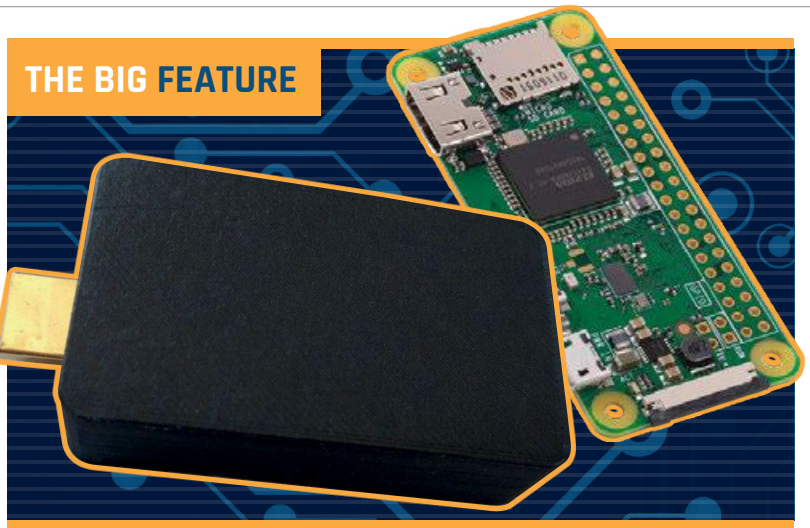
Upgrades sent to space **10**

SUSE ON PI



Enterprise-class OS now available **13**

Contents



THE BIG FEATURE

PI ZERO TV STICK

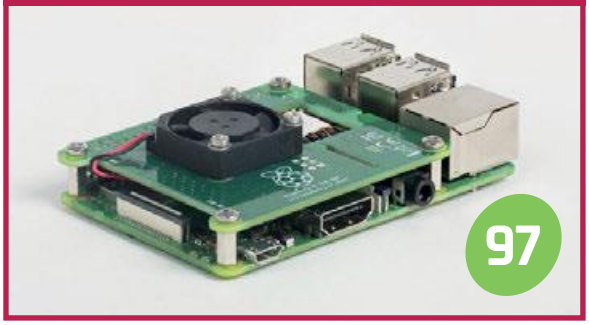
The most portable Pi computer ever!

62

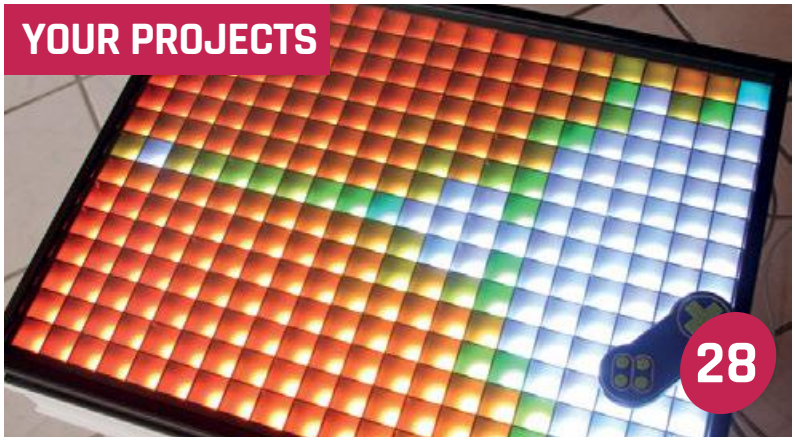
WIN!

Raspberry Pi

10 RASPBERRY PI 3B+ & PoE HATS



97



YOUR PROJECTS

28

LED MATRIX

An amazing table made of giant pixels

KODAK BROWNIE CAMERA **30**
Upcycling a classic camera with a Pi

EMBROIDERY MACHINE **32**
Using a Raspberry Pi to power-up sewing

PANNING FOR GOLD **34**
How a Pi is used as a timer for a unique sport

SOFI ROBOT FISH **36**
We look at an incredible robo-fish, powered by Pi

REGULARS

> NEWS **06**
> TECHNICAL FAQ **58**
> BOOK REVIEWS **80**
> FINAL WORD **98**

COMMUNITY

> UNIVERSITY LAB INTERVIEW **84**
Monitoring lab usage with a Raspberry Pi and IoT

> THE MONTH IN RASPBERRY PI **86**
What else happened this month in the world of Pi?

> CICELY DAY PROFILE **90**
A Californian teacher with a magic touch

> UPCOMING EVENTS **92**
What Jams and other events are going on?

> YOUR LETTERS **94**
Emulation, kernels, and more in your letters

REVIEWS

> RASPAD **70**
> GETTING YOU STARTED KIT **72**
> PUREAUDIO ARRAY MDK **74**
> NOODLE PI **76**
> WIDE INPUT SHIM **78**

NEW GOOGLE AIY PROJECTS KITS

Updated Vision and Voice Kits contain a Pi Zero WH and microSD card; new Android app revealed

Google has launched updated versions of its wildly successful AIY Vision and AIY Voice Kits. The new kits include a Raspberry Pi Zero WH, a pre-provisioned microSD card, and – with the Vision Kit V1.2 – a Raspberry Pi Camera Module V2.

"Everything you need to get started is right there in the box," says Billy Rutledge, Google's Director of AIY Projects.

The aim is to make setup easier. Users no longer need to buy additional parts, or download the software image. Having all the parts in one place makes setup faster, and it ensures newcomers have all they need.

Above The new AIY Vision Kit V1.2 even includes the camera

Pi inside

"The updates allow Google to continue to give makers an easy way to integrate AI in their products and creatively use AI to solve their own challenges," Billy explains.

The aim of AIY Projects is to make it cheaper and easier to add AI processing to a project. "We knew from our research that even though makers are interested in AI, many felt that adding it to their projects was too difficult or required expensive hardware," continues Billy.

Voice and image recognition were "the two biggest areas of interest

Above The new AIY kits include a Raspberry Pi Zero WH, so everything you need is in this one box

> The new kits move away from using a Raspberry Pi 3 towards the smaller Pi Zero form factor

for the community", he goes on to tell us, so that's where the AIY Projects team started. Both the AIY Vision and AIY Voice Kits have "sold out of inventory with each production run," Billy reveals.

The new kits move away from using a Raspberry Pi 3 towards the smaller Pi Zero form factor.

As we are told, "we designed our first version of Voice Kit around the Raspberry Pi 3," but many AIY builds used "the lower-cost Raspberry Pi Zero, even with some of the Google Assistant SDK features being limited for the ARM v6 instruction set."

"We thought it would be a great opportunity to showcase the performance and flexibility of the smallest Raspberry Pi. Not to mention lower the cost of the kit."

Top of the class

The AIY Projects kits were huge hits amongst the maker community, but they also proved pretty popular in classroom environments, to boot.

"We're seeing continued demand for the kits," explains Billy, "especially from the STEM audience where parents and teachers alike have found the products to be great tools for the classroom."

The new AIY kits are on sale at US retailer Target now, which is also now stocking the Raspberry Pi itself (handy for our US readers). The new kits will be coming to the UK "this summer", Billy concludes.

The AIY Voice Kit V2 costs $50, the AIY Vision Kit V1.2 $90; UK pricing hasn't yet been confirmed, but we hope to have more information on pricing and availability soon. Sign up for our newsletter (**magpi.cc/email-me**) to get more info on UK availability of the AIY Projects kits.

ANDROID APP

Google has introduced a new AIY Projects companion app for Android devices. The app makes wireless setup and configuration an absolute snap.

The Android app is available alongside the launch of the new kits from the Google Play store (magpi.cc/hnnWbZ).

"Don't worry, the kits still work with monitor, keyboard, and mouse," says Billy. So you won't need an Android smartphone to run the AIY Projects kits. Google is also working on iOS and Chrome companion apps, which should be coming along soon.

To help users, the AIY Projects website (aiyprojects.withgoogle.com) has been refreshed with improved documentation. The aim is to make it easier for young makers to get started and learn as they build.

Also new is an AIY Models area, showcasing a collection of neural networks designed to work with AIY kits.

FORMULA PI CHAMPION ANNOUNCED

Pi-controlled autonomous racing

The latest season of Formula Pi has come to a glorious – if slightly chaotic – end, with team lambda.p.racing the clear winner.

For those that haven't seen a live-streamed Formula Pi race, it's an autonomous racing challenge where entrants submit code to run on standardised MonsterBorg robots.

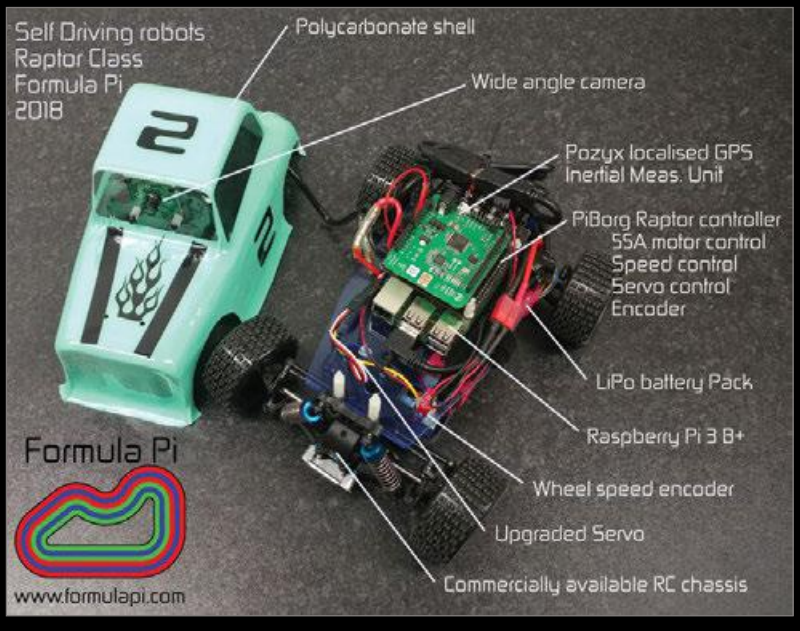
As Formula Pi race director Timothy Freeburn explains, this means competitors "do not require a large budget or their own robot to compete, with the cost of entry being just over $50 (£37)." Formula Pi even provides basic example code to get you started – see **formulapi.com**.

Left The next season of Formula Pi will feature the new, faster Raptor class

Winning code

This season of Formula Pi saw some big changes. "We added challenge races," says Timothy, "where two competitors raced head-to-head whilst having to avoid stationary robots on the track."

While these "had a mixed reception," Timothy adds that "the intention was to make sure competitors thought about adding avoidance code to their entry."



Above You can paint your Raptor's shell, as well as write its controlling code

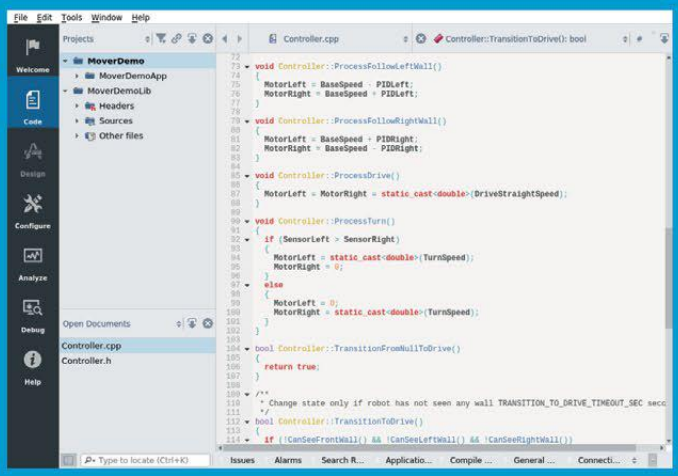
This seemed to pay off, as the final race – which you can see at **magpi.cc/cwmmPt** – featured a huge crash. Timothy tells us "lambda.p had a bespoke avoidance algorithm which worked quite well... there was probably a small amount of luck involved as well."

New season, new robots

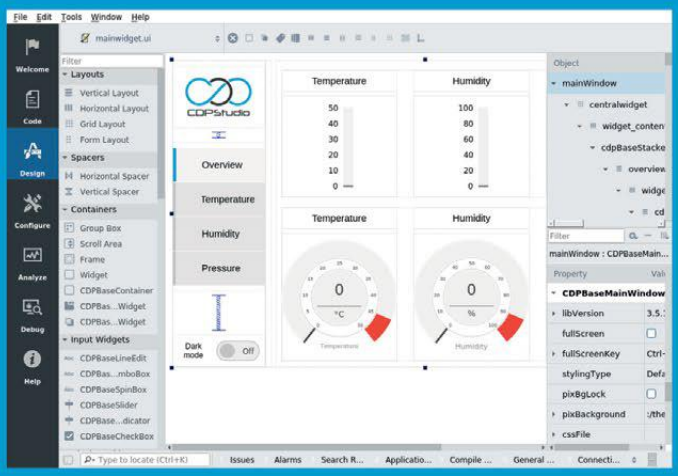
Registration for the Summer 2018 series is already open, and there's a new category of robot for experienced Formula Pi racers. The new Raptor class is by invitation only "for the moment," Timothy reveals, because Raptors are "much faster than the MonsterBorgs and have a Pozyx localised GPS, inertial measurement unit (IMU), and a wheel speed encoder on board."

The current MonsterBorg category will continue as a parallel formula; entry is open until 31 May – see **magpi.cc/loKCZz**.

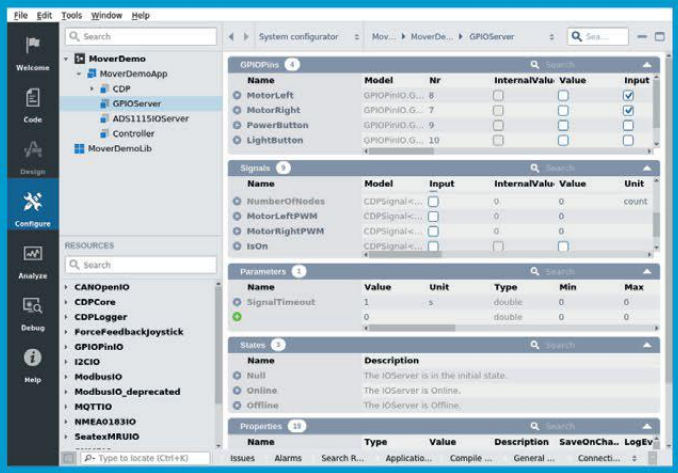
If you like the sound of Formula Pi and want to give it more support, there is an opening for a major sponsor. Head over to **magpi.cc/uDaLYs** for more details.



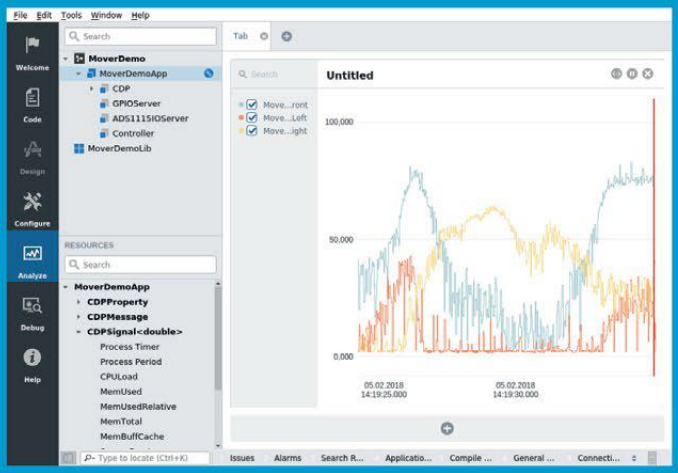
Code



Design



Configure



Analyze

Now free for home projects

A professional control system development tool

CDP Studio is a development platform for industrial control systems, now coming with a free version for non-commercial use. The system can run on a Raspberry Pi, supports C++, open source libraries and has a large feature toolbox including GPIO, I2C and MQTT. Its built in GUI design tool and features lets you code less and do more.

Free download on www.cdpstudio.com

CDP Technologies AS
Nedre Strandgate 29
P.O. Box 144
NO-6001 Ålesund, Norway

Tel: +47 990 80 900
info@cdptech.com
www.cdpstudio.com

CDPStudio

WEBOS COMES TO RASPBERRY PI

webOS OSE open source and 'optimised for Pi'

The webOS operating system – developed by Palm, acquired by HP, licensed to LG Electronics, and sold to Qualcomm – now has an official open-source version optimised for the Raspberry Pi.

The new OS has been launched by LG but is an independent project, with its own website and name: webOS Open Source Edition, or webOS OSE for short.

The initial offering is fairly basic, with only a few apps built in. However, you can develop your own apps using IoTivity and iotivity-node.

Despite the v1.0 release number, even getting webOS onto your Raspberry Pi is tricky. You need to be running Ubuntu 14.04 LTS 64-bit to compile the source code, for example, as no pre-built image has been issued.

This is despite the Raspberry Pi 3B being the only recommended system for webOS OSE. Helpfully, forum user Vipeax is hosting a pre-built image for download – see **magpi.cc/sxMNvw**. Also, visit **webosose.org** for details of webOS OSE, and how to contribute.

Image courtesy of: webOS OSE, webosose.org

Above The latest new incarnation of webOS is open source and optimised for the Raspberry Pi 3

ASTRO PI UPGRADES

Now with wireless and NoIR filters

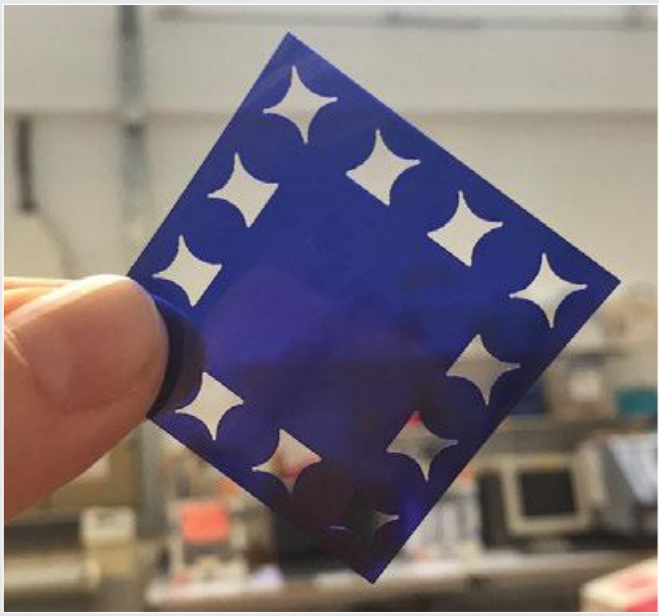
The two Raspberry Pis hurtling around the Earth at 17 500 mph aboard the International Space Station have been upgraded.

The 'payload' of upgrades was launched into space on 21 March on a Russian Soyuz MS-08 (54S) rocket crewed by Oleg Artemyev, Andrew Feustel, and Ricky Arnold.

Two USB dongles add wireless connectivity to the two Astro Pi units, which are based on the Raspberry Pi 1 B+ and have therefore relied on Ethernet networking. Now the Astro Pi units can participate in experiments in every area of the ISS, not just the Columbus Module.

Alongside the two wireless dongles are four 32GB SD cards "so that future Astro Pi code will need to command fewer windows to download earth observatory imagery to the code", Dave Honess, Raspberry Pi Foundation education resource engineer (now with the European Space Agency), explains.

Lastly, the upgrade package contained five NoIR filters to allow the Astro Pi cameras to see into the infrared portion of the light spectrum.



Above The new Astro Pi upgrades bring greater freedom and possibilities for Raspberry Pi-based space science

CODE CLUB AT AGRIHACK

Solving farming problems with a Raspberry Pi

Agrihack 2018 brought together over 35 organisations to create seven different solutions to three tricky problems.

Code Club Australia's national programme manager Nicola Curnow explains, "We help out on the first day of the event, the kids' day. And the second day and third day is a Hackathon for adults."

> "In Australia, connectivity is a big issue"

Nicola also explains that "in Australia, connectivity is a big issue, [so] almost everything was offline." The Code Club day mainly used Scratch, but "a small group of advanced kids... did a Raspberry Pi / Sense HAT program about weather."

"The kids visit a farm in the morning," Nicola continues, "do a design thinking challenge with the sponsor of the event, and then participate in our coding workshops."

The Hackathon challenges were: reducing stored grain spoilage from high moisture, measuring the live weight of meat chickens to prevent overfeeding, and assisting the handover of farms and fisheries from the older to the younger generation. You can see the winning solutions at **magpi.cc/BSrjpg**.

Below Code Club Australia brought kids and farmers together to see how tech can support food production – this year's challenge was to find sustainable ways to reuse milk containers

NOW TRENDING

The stories we shared that flew around the world

INSTALL UBUNTU MATE ON YOUR PI

magpi.cc/vTeNPF

While many of us use Raspbian for our Raspberry Pi needs, there is another Debian-based distro to choose. With commercial support from Canonical Ltd, Ubuntu MATE is a worthy alternative for faster Raspberry Pi devices.

PISWITCH - A DIY NINTENDO SWITCH

magpi.cc/dvjWNH

Christopher Foote invites you to make your own Nintendo Switch games console, using retro game emulator RetroPie, a Raspberry Pi, a touchscreen, and a pair of Nintendo Joy-Con controllers.

RASPBERRY PI 3B+ LAUNCHED

magpi.cc/ooUTzs

The new Raspberry Pi 3B+ packs a lot of upgrades into the same form factor and same price as the original Pi 3B. The wireless is faster and more reliable, there's Gigabit Ethernet, and the processor runs faster.

COMPUTE MODULE MOTHERBOARD

Resin.io releases Project Fin

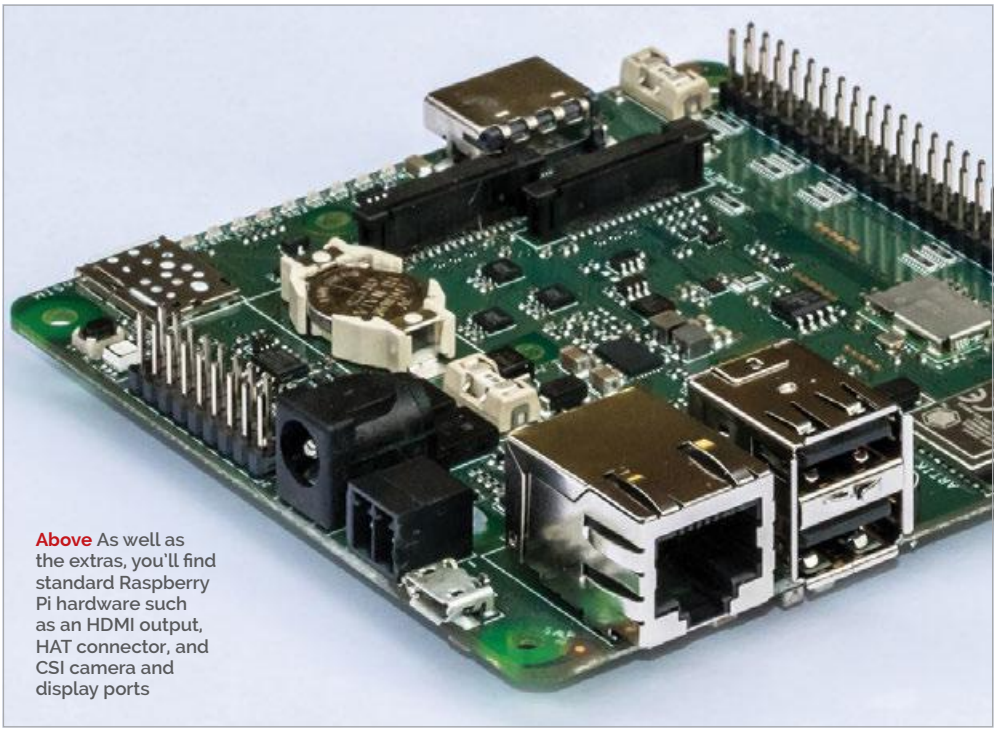
Better known as an IoT software firm, resin.io has launched its first hardware – a Raspberry Pi Compute Module carrier board called Project Fin.

A carrier board is just like a motherboard for a desktop PC, with the Compute Module (CM) acting like a plug-in processor. Just as with a typical motherboard, different carrier boards offer different features.

The Raspberry Pi Foundation offers the Compute Module IO Board V3 as part of a development kit (see **magpi.cc/Wxymor**), but resin.io's Project Fin has a few extra features on offer.

For example, Project Fin has dual-band (2.4GHz and 5GHz) WiFi and Bluetooth 4.2, plus a connector for an external WiFi or Bluetooth antenna. The Bluetooth radio can even operate when the CM is powered down. The CM can be powered down or up by code, accessing Project Fin's on-board real-time clock (RTC) chip.

A mini PCI-Express slot enables the attachment of standard PC peripherals – a cellular modem, for example, which could work with the built-in SIM card slot. There's also more robust power delivery, allowing input voltages of 6–30V (or 5V via the standard HAT connector).



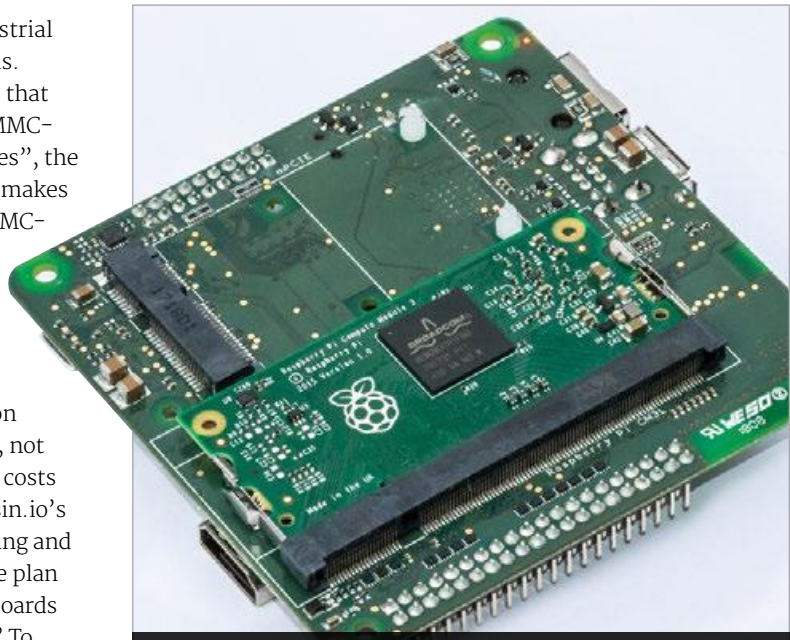
Above As well as the extras, you'll find standard Raspberry Pi hardware such as an HDMI output, HAT connector, and CSI camera and display ports

Hardened design

Carlo Curinga, resin.io's Director of Technology Operations (Europe), also explains that Project Fin has "one of the best available eMMCs. Paired with strong power circuitry, this ensures fast and reliable operations… we avoid all the pitfalls of SD cards," such as relatively short (in industrial terms) read-write lifespans.

Although Carlo confirms that while "Fin does support eMMC-populated Compute Modules", the on-board eMMC obviously makes Project Fin ideal for the eMMC-less Compute Module 3 Lite (CM3L), based on the Raspberry Pi 3B.

Project Fin boards will offer 8–64GB of on-board eMMC, with the 8GB version given a target price of $129, not including the CM3L (which costs $25–$30). Alison Davis, resin.io's director of product marketing and strategy, confirms that "we plan to make the first batch of boards available later this spring." To keep up to date, email your interest to **fin@resin.io**.



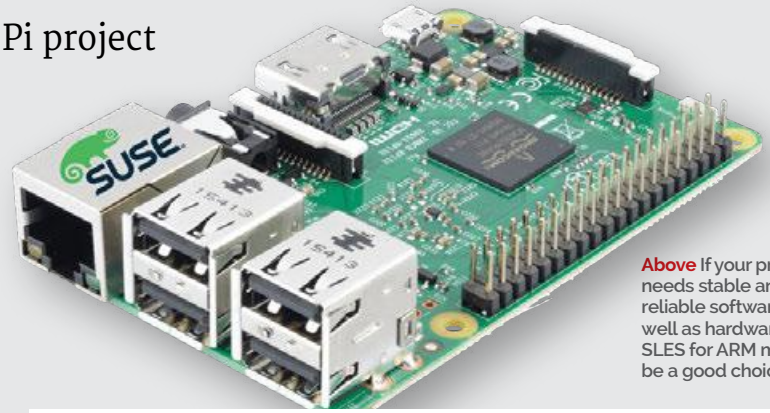
Essentially a motherboard for a Raspberry Pi Compute Module, Project Fin is packed with extra hardware and capabilities

SUSE LINUX COMES TO PI

Enterprise-class OS for your crucial Pi project

SUSE Linux has released a version of its super-robust server OS for the Raspberry Pi, called SUSE Linux Enterprise Server for ARM.

As Jay Kruemcke, Product Manager for SUSE Linux Enterprise Server for ARM explains, "SUSE had multiple customers implementing industrial monitoring solutions based on the Raspberry Pi that wanted to use a Linux OS with the same support, security, and reliability that they already used in their data centre."



Above If your project needs stable and reliable software as well as hardware, SLES for ARM might be a good choice

> "SLES for ARM can be used for all kinds of systems"

The new OS follows a demo of SUSE Linux running on a Raspberry Pi at SUSECON 2016, which gained "a tremendous amount of interest" and generated "thousands of downloads in just the first few days," according to a SUSE Linux statement.

Jay says the main challenge in releasing a fully stable version of SUSE Linux Enterprise Server (SLES) for ARM was timing: "We had to do the Raspberry Pi support after SLES 12 SP3 was already available." SLES 12 SP3 was released on 6 March, with SLES for ARM launched only 20 days later.

SLES for ARM can be used for all kinds of systems – Jay tells us it's already in use by some of SUSE's early customers for industrial monitoring.

"We intend to make subscriptions for SLES on ARM available... within the next few weeks," Jay confirms, while anyone with a SUSE Linux customer account can download the 60-day trial of SLES for ARM from **magpi.cc/whHWgx**.

Above SUSE Linux Enterprise Server for ARM is an enterprise-grade server OS optimised for the Raspberry Pi 3B

pi-top

Inspiring inventors and creators to seek the skills of tomorrow and create their future, today.

The modular laptop with sliding keyboard

8HR BATTERY LIFE

14" FULL HD 1080P SCREEN

180° HINGE

CUSTOM PASSIVE COOLING BRIDGE

MODULAR RAIL

pi-top

Colors
Raspberry Pi 3 optional

AWESOME INVENTOR'S KIT INCLUDED

20+ projects to explore

Explore beyond the screen and keyboard by creating with the all-new **pi-top** modular laptop.

Get started with 20+ inventions in the inventor's guide booklet. There are 3 inventor's journeys - Smart Robot, Music Maker and Space Race.

The modular desktop

14" FULL HD 1080P SCREEN

MODULAR RAIL

ADJUSTABLE VIEWING ANGLES

pi-topCEED

Colors
Raspberry Pi 3 optional

pi-topCEED is the plug & play modular desktop. It's the easiest way to use your Raspberry Pi. We've put what you love about our flagship laptop in a slimmer form factor. Join hundreds of code clubs and classrooms using **pi-top**CEED as their solution to Computer Science and STEAM-based learning.

pi-topPROTO

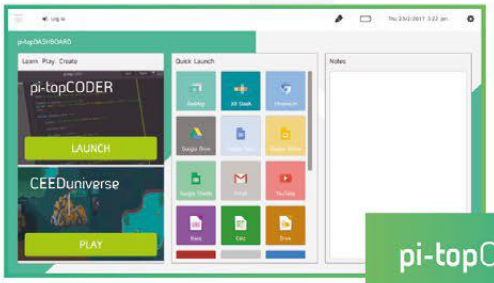
pi-topSPEAKER

pi-topPULSE

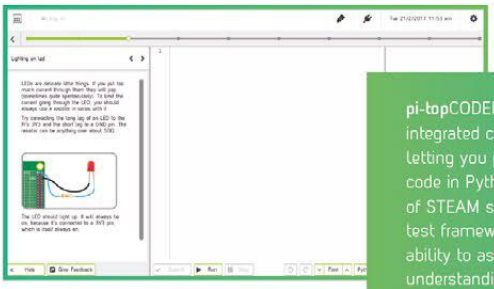
Modular **Accessories**

pi-top is an award-winning ecosystem designed to make experimenting, coding and building electronics, simple, affordable and fun. **pi-top**OS is here to guide you through the world of making!

The OCR* endorsed **pi-top**OS (Operating system) platform comes pre-installed on the SD card shipped with every unit. **pi-top**OS software suite lets you - browse the web, - check emails, - create and edit Microsoft Office compatible files. Gain access to dozens of hands-on learning lesson plans with **pi-top**CODER and have fun learning to code with CEED**universe**!



pi-topOS



pi-topCODER has a fully integrated coding environment letting you program hardware, code in Python and learn lots of STEAM skills! Our integrated test framework gives you the ability to assess your own understanding as you learn.

CEED**universe**

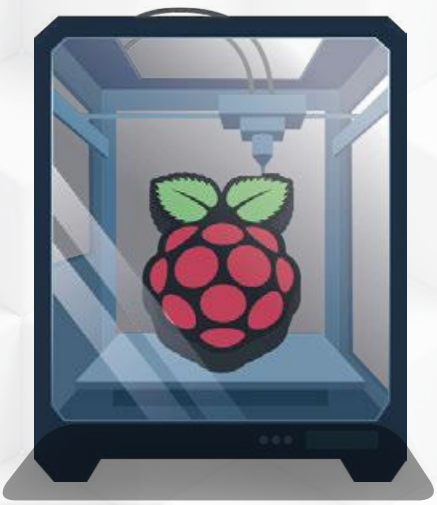
Learn programming concepts through our minigames, for example, learn problem decomposition by solving visual programming puzzles.

AFFORDABLE 3D PRINTING WITH RASPBERRY PI

GET INTO 3D PRINTING USING RASPBERRY PI AND A LOW-COST 3D PRINTER

Maybe you've noticed, but 3D printing has become a whole new board game in the last few months. You can now pick up a 3D printer from iMakr for just £99. And there are dozens of different 3D printer models that can happily be picked up for around £150.

Combining a low-cost 3D printer with the well-engineered Raspberry Pi is a very tempting proposition.



Affordable 3D printers may lack frills such as touchscreen displays, but the Raspberry Pi adds more than enough computing power to transform an entry-level model into a professional piece of kit.

Hooking a Raspberry Pi up to a 3D printer makes the whole process easier, not more complex. Custom-built software such as OctoPrint handles most of the 3D printing process for you. You can even buy, or build, touch-enabled kits like the Raspberry Pi-powered AstroBox to instantly transform a 3D printer kit into a ready-to-go 3D print centre.

There has never been a better time to get into 3D printing. It's finally affordable and the Raspberry Pi makes it fun.

In this feature we're going to cover all the basics, from buying a low-cost 3D printer, to installing the software on your Raspberry Pi, downloading 3D printer files, and printing your own media player case for a Raspberry Pi.

01. STEPPER MOTORS

The extruder section and bed are moved using stepper motors. These provide precise motion by rotating in increments (typically 200 steps per rotation). These adjust the position of the part in three-dimensional space while the nozzle prints out more heated filament. The positioning of the parts is controlled using G-code, which is what our Raspberry Pi will be helping with.

02. COOLING FAN

All good 3D printers come with a cooling fan and it's a must-have feature. The cooling fan points to either the nozzle or the build area and actively cools the plastic after it's been shaped by the nozzle. This crisps up the project quickly, reducing squidgy edges and improving any bridged sections.

03. EXTRUDER AND NOZZLE

The part that builds the 3D project up is made of three parts: the extruder, hot end, and nozzle. The extruder feeds the filament into the hot end, which (as the name suggests) heats it up. It's then squeezed through the nozzle and placed on the bed. The nozzle and bed move around to build up a 3D-printed part, one layer at a time. The nozzles can come in different sizes, with smaller nozzles resulting in a smoother finish and larger nozzles providing faster printing.

04. BUILD AREA

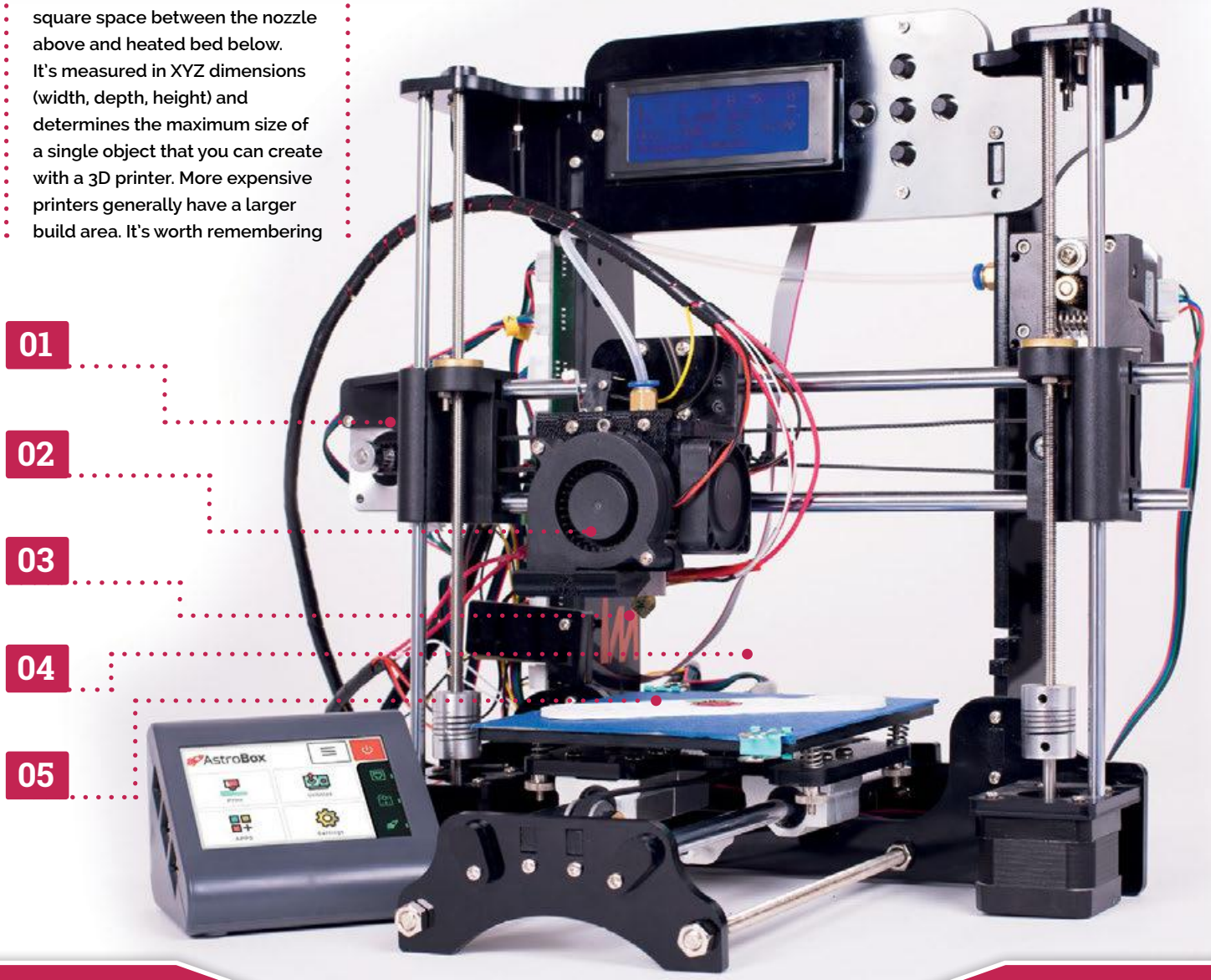
The build area (or volume) is the square space between the nozzle above and heated bed below. It's measured in XYZ dimensions (width, depth, height) and determines the maximum size of a single object that you can create with a 3D printer. More expensive printers generally have a larger build area. It's worth remembering that bigger print jobs can be split into smaller parts and combined afterwards.

05. BED

The bed is the flat area that your 3D-printed project is built up on. More expensive 3D printers come with heated beds. These enable them to work with high-temperature filaments. But our entry-level models come with regular unheated beds which limit them to PLA plastics (see the 'Filament' box). The bed must be level before 3D printing starts. More expensive printers have a system for automatically levelling the bed, but entry-level models are often calibrated by hand.

FILAMENT

The stuff that your 3D-printed projects are made of. Filament is a thread-like thermoplastic that is fed into the 3D printer, heated, and shaped layer-by-layer into an object. There are two standard diameters: 1.75 and 3.00 mm. An entry-level printer will probably be using 1.75 mm filament, but be sure to check the technical specifications for your particular model. There are many different types of filament available, with PLA (polylactic acid) being a popular, and arguably the best, choice for beginners.



BUYING A 3D PRINTER

3D PRINTING IS NOW FOR EVERYONE THANKS TO RASPBERRY PI

PJ EVANS

PJ bought his 3D printer in December 2017. Since then he's littered an otherwise clean and tidy house with things his wife describes as 'Christmas cracker tat'. He remains hopeful.

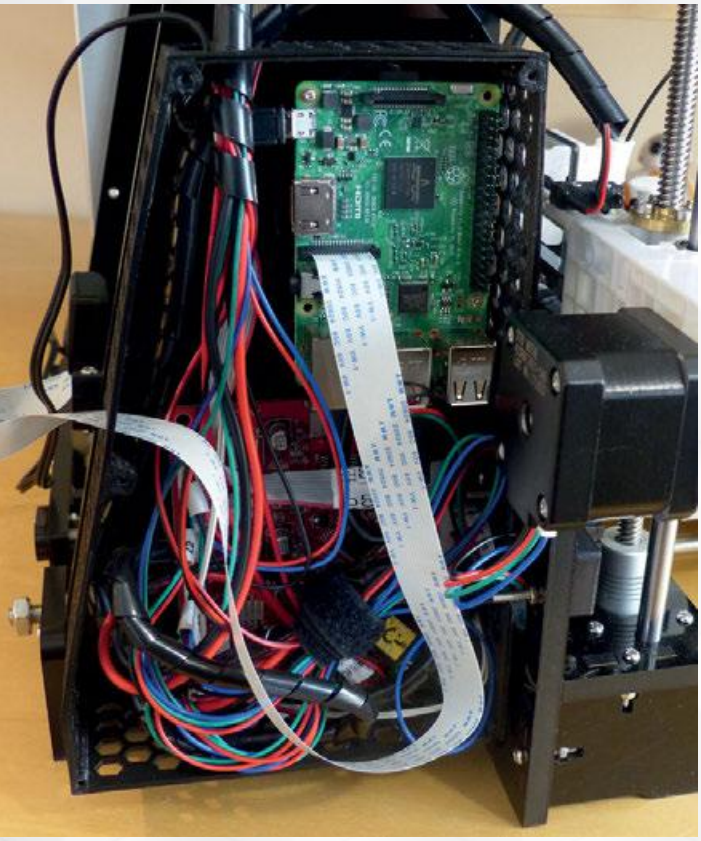
mrpjevans.com
twitter.com/mrpjevans

It was less than ten years ago that the first consumer-level 3D printer, the MakerBot 'Cupcake', went on sale.

Sold as a kit for $750, it could print small objects up to 100 × 100 × 130 mm. Impressive, but strictly for the enthusiast. Now, just a few years later, affordable and vastly more capable printers are commonplace. Thanks to the open-source community behind the original printers, and the advances in manufacturing and techniques, 3D printing is well within the grasp of anyone curious to know what happens when plastic goes wrong. We're kidding. Well, mostly.

In 2005, a project began in Bath, England, to build an affordable 3D printer that could replicate its own parts as much as possible. This 'self-printing' printer was called the RepRap. It was so clean and elegant that enthusiasts took the open-source design and started to improve it, creating a family of printers based around the original RepRap design that are not only affordable but also very capable of printing impressive 3D objects.

Better still, when you add a Raspberry Pi into the mix, you get a whole new level of control and print quality.



Below You can permanently install your Raspberry Pi in a self-printed enclosure. Most i3 3D printers connects to a Raspberry Pi using a simple USB connection. The Pi can then be motunted within a printable enclosure

DIVING IN

Your author finally took the plunge late in 2017 and purchased a printer based on the popular Prusa i3 family of printers, which themselves are an evolution of the original RepRap. What I received would have made any hardened DIY veteran shiver. Inside the three trays were a mass of parts, wires, and very little documentation.

I wouldn't have stood a chance if it hadn't been for an exhaustive YouTube video that detailed the entire construction process. All in all, it took about eight hours to assemble the printer, yet I soon learnt that this was just the beginning.

I3 PRINTERS

Many 3D printers are based on the popular Prusa i3 MK2 and are well priced at around £200. The basic kits are good enough for learning 3D printing, but soon you'll be improving your printer with the large number of modifications available.

prusa3d.com

THREE **AFFORDABLE** PRINTERS

£99

IMAKR STARTT

An amazing low-cost printer particularly suitable for new makers, smaller budgets or tight space requirements. It has a small build volume, but matches the resolution of the bigger players.
magpi.cc/AjfnMV

WANHAO DUPLICATOR I3 MINI

The Wanhao Duplicator i3 Mini arrives fully assembled and ready to print. With a 120 × 135 × 100 mm build volume, it's perfect for smaller projects and is fully compatible with software for the i3 family.
magpi.cc/uYWeFh

£199

£899 ASSEMBLED
£699 KIT

PRUSA I3 MK3

Josef Prusa's printers are considered the gold standard for consumer 3D printing. The new i3 MK3 boasts many impressive upgrades. If quality and reliability are your priorities, look no further.
prusa3d.com

Once assembled, the printer's 'head' – the extruder – required careful calibration. Again, YouTube videos helped me slowly measure the extruder's height above the printing base, which needed to be around 0.2 mm. Yes, one-fifth of a single millimetre.

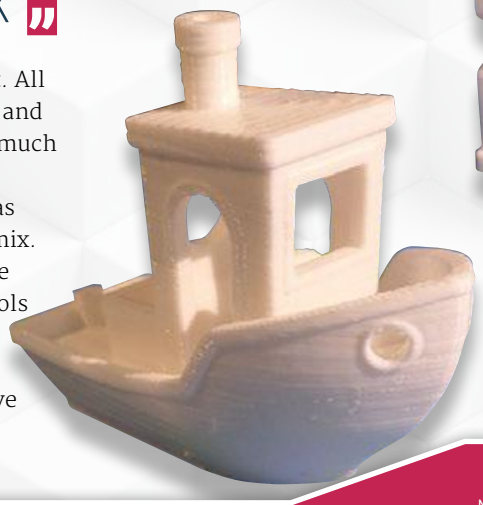
3D printing has a big learning curve, but it's a fun one and very rewarding. As you start to understand the mechanics of your printer, you'll receive instant praise in the form of successful prints. Realising my i3 printer was a very basic kit, I discovered it could be improved by printing out its own modifications. There's a vast array of freely downloadable parts to enhance your investment. Better still, this is great practice for your early days in 3D printing. Make all your mistakes (there will be many, but they look hilarious) whilst making your printer better.

PRINTING PARTS

Within a week or two, my printer had stabilisers (not the bike variety), an improved filament feed, a button for the feed release and an improved cooling vent. All of these cost pennies to print and made my new favourite toy a much better printer.

> "THE BIGGEST LEAP FORWARD WAS ADDING A RASPBERRY PI TO THE MIX"

The biggest leap forward was adding a Raspberry Pi to the mix. The superb OctoPrint software package takes the basic controls of a standalone printer and turns it into something that just a few years ago would have cost thousands to purchase. Sure it requires patience; there's a lot of trial and error. Then comes a day when you're reliably printing amazing stuff first time. You can then venture into the world of designing your own models for printing. Recent software such as TinkerCAD makes light work of designing complex objects and sending them to your printer. From your imagination to reality.



Above The 'Benchy' boat and Cali Cat are stress tests for your printer to check calibration, adhesion and temperature. You get some cute prints as a reward

YOU'LL NEED

- Prusa i3-compatible 3D printer
- Cura settings file. This is normally provided with your printer
- 1.75 mm PLA filament (we're going to use two different colours)
- USB cable
- OctoPi image **magpi.cc/MbziUP**
- 2GB or larger microSD card (for the OctoPi image)
- Case files **magpi.cc/sycQkZ**
- Pi Camera Module (optional)

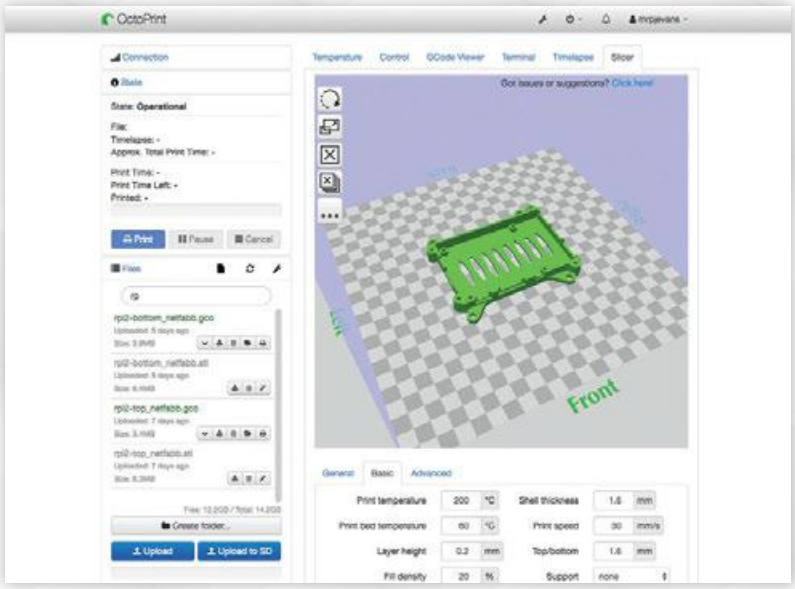
SET UP A 3D PRINTER WITH RASPBERRY PI

GET YOUR RASPBERRY PI TO PRINT ITS OWN AMAZING CASE WITH A 3D PRINTER AND OCTOPRINT

One of the most popular accessories for the Raspberry Pi is a case. There are countless options available, but some of the cleverer designs can be 3D-printed. Not only can you choose your type and colour, but also make modifications if you need to.

We're going to print a case that features mini-VESA mounts for attaching your Pi to the back of a television; perfect for a media centre project. Using a Raspberry Pi running OctoPrint – a freely available web-based control centre – we'll get easier control, remote monitoring, and better results.

We're going to achieve two things in this tutorial. First we're going to improve a standalone 3D printer by adding OctoPrint running on a Raspberry Pi. Then, we'll use this setup to prepare and print a nice new case for the Raspberry Pi.



Right OctoPrint's built-in slicing makes 3D printing easier than ever before

Most 3D printers can run in a standalone mode, but this often involves fiddly menus on tiny LCD screens and many useful options being oversimplified. OctoPrint provides a web-based interface for your printer, making for a much friendlier experience. Better still, OctoPrint provides support for a Raspberry Pi Camera Module so you can monitor the process and create time-lapse videos.

Probably the most useful feature is the inclusion of a 'slicer', the critical piece of software that converts a 3D model (an STL file) to something a layer-based 3D printer can understand. The language of 'Fused Filament Fabrication' printers is G-code, and slicing software such as Cura and Slic3r generate this code from STL files. OctoPrint allows you to go straight from an STL file to printing by slicing for you.

GET OCTOPI

Start by using your favourite tool – such as Etcher – to burn your downloaded OctoPi image to a microSD card (yes, it's called OctoPi but should not be confused with

This print is in two parts, so you can change colours if you like
This base fits a Mini VESA mount. An alternative full-size mount is also provided
Secure the case with four bolts (M2.5 × 12mm)

the OctaPi Raspberry Pi clustering project). This Raspbian Jessie-based image gives you everything you need to get started.

Use Etcher to write the image to a microSD card. See our 'How to use Ether' tutorial if you don't know how to write image files to SD cards (**magpi.cc/etcher**).

If you plan on using wireless LAN, instead of wired Ethernet, have a look in the **boot** folder on the card (you may need to remount it) and open the file **octopi-network.txt**. In there are instructions on how to set your network's SSID and password. After editing the file, save it back to the SD card, unmount it, and insert it into your Raspberry Pi. It will now boot and connect to your network first time.

Power up your Pi and give it a minute or so to get a network connection. If you want to see what's happening, you can hook up a monitor and keyboard, but it's not necessary. You should then be able to access **http://octopi.local** from any web browser on your network. If you can't access the site, you made need to discover the Pi's IP address by logging in and entering **`ifconfig wlan0`** at the console, then go to **http://<ip address>** instead. On first visit, OctoPrint will go into 'Setup' mode.

SET UP CURA

You will be asked for a copy of your Cura profile. Cura is one of the most popular 'slicers' available and nearly every printer will come with a Cura profile in the form of an INI (.ini extension) file. If you are already using Cura, you can export the profile directly from the application.

Once the setup is complete, there's just one final step. OctoPrint supports plug-ins to enhance its operation. The included slicing software is good but basic. Fortunately, a plug-in exists to give you fine control over your printing. Click on the wrench on the navigation bar, then Plugin Manager which allows you to download and install plug-ins. If you find the repository is unavailable, go to Software Update in Settings and update OctoPrint to the latest version. Now search for 'Full-featured Slicer' and install it. This plug-in by Kenneth Jiang gives enhanced access to Cura's controls and is essential for great prints direct from STL models.

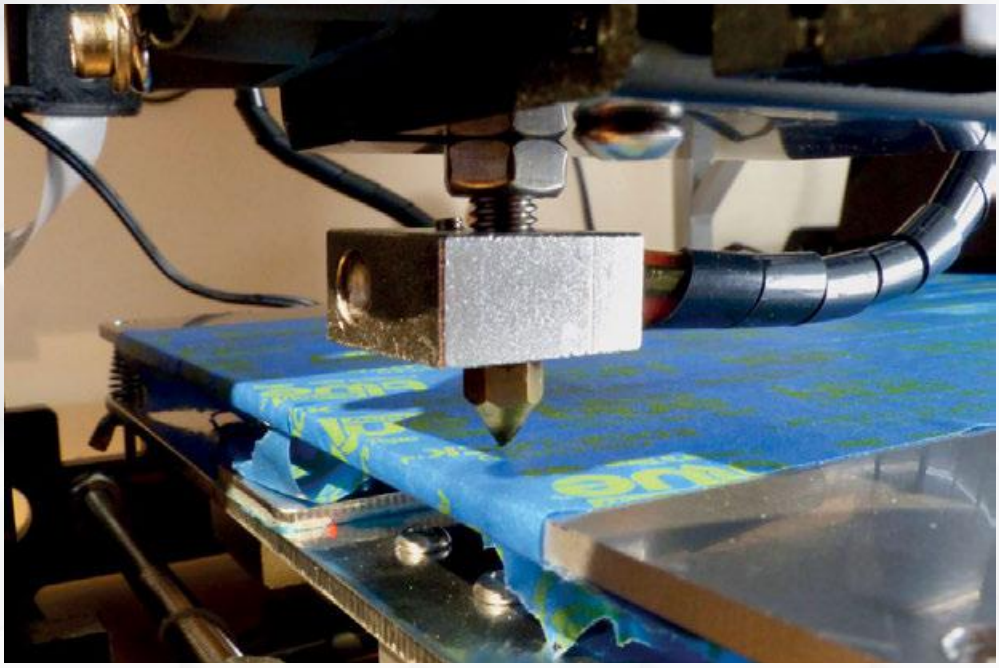
Now OctoPrint is ready to go. It's time to start up your printer and connect using the USB cable. On the homepage, click Connect and wait

ASTROBOX

You can take things to another level by turning a Raspberry Pi into an AstroBox Gateway.

You can buy the parts and set it up yourself, or buy a pre-built model direct from AstroPrint. The £100 AstroBox Gateway is a Raspberry Pi-based 'link' between your printer and the cloud, while the AstroBox Touch (£200) is a standalone device that provides advanced control via a 3.5-inch touchscreen (see page 17). Both offer control and monitoring of your printer from anywhere in the world.

astroprint.com



Above Levelling is one of the most important parts of successful 3D printing. Too high, the filament won't stick. Too low, you risk damage to your printer

a few seconds. You'll start to see temperature readings coming in.

DOWNLOADING FILES

For this tutorial we'll use a file from Thingiverse, a site offering thousands of 3D files covering models, games, jewellery, cosplay, tools, printer modifications, and a range of cool Raspberry Pi cases.

Another site, MyMiniFactory, is also rapidly increasing in popularity. This offers a different range of items, including two fantastic projects you'll find near the end of this feature (the jRobot and Swiss army SD card). It's well worth bookmarking both sites.

We're going to print a file from Thingiverse designed by the mysteriously named 'M-P'. Item number 922740 is a two-part Raspberry Pi case that fits the Pi 2 and 3. Helpfully, several 'tops' and 'bottoms' are provided. The 'tops' feature different fan designs, while the 'bottoms' fit different VESA mounts. You can choose which combination to print. Once you've unzipped the download, you'll find the all-important STL models in files. I've chosen **rpi2-top_netfabb.stl** and **rpi2-bottom_netfabb.stl**.

Back in OctoPrint, upload the two STL files by simply dragging them onto the webpage or by clicking Upload. Then click the little magic wand and you'll be taken to the slicing screen. This is where you can control how the model is printed.

SLICING THE FILE

There are some important considerations before we start to print the file – chiefly temperatures used, supports, and adhesion.

The manufacturer will recommend a temperature for the extruder and the printing bed. It's generally a good idea to follow these guidelines and only make tweaks if you see signs of poor layering or warping. We generally print at 200°C on a 60°C bed, but factors such as ambient temperature, air currents, and choice of filament will all have an effect on the final print quality. If you need to change these settings, try adjusting in 0.5°C steps.

If a model is going to have an argument with gravity, you may need supports. The slicer will automatically add very thin supporting material so the object won't fall down. If you imagine trying to print a bridge without supports, the printer will be laying plastic down in thin air and then you have spaghetti. Adding supports will place easily removable columns along the bridge to support it as it prints. They simply snap off when printing is complete.

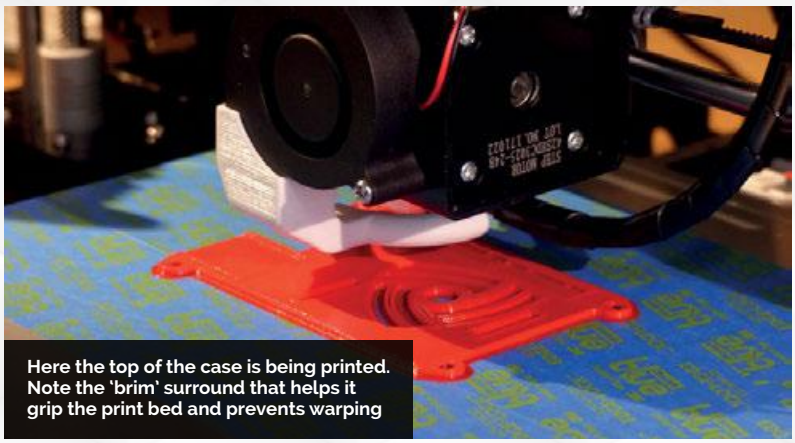
Adhesion is another important consideration. The first layer of material is the most important and a deciding factor in whether a print will be successful. With no adhesion setting, the print will be placed down as is, typically with a 'skirt' around it to ensure the filament is smoothly extruding before print starts. A 'brim' will print a very thin layer around the edges of the model to help it grip the printing bed. For more complex prints, a 'raft' is a low-resolution base on top of which the model is printed. This generally gives best adhesion and reduces warping. After printing, the raft easily peels off. For our case, we're going to use a brim.

The case is already correctly sized so click 'Slice It!' and OctoPrint will generate the G-code instructions for your printer. They'll appear as a new file with a '.gco' extension.

FILAMENT TYPES

Trying out different filaments is part of the fun of 3D printing. The most popular is PLA as it melts at lower temperatures and is well balanced between strength and flexibility. If you need a stronger material, ABS is widely supported but is more difficult to work with. Many others are available, including wood-effect and PLA with metallic powder that rusts.

filamentguide.net



Here the top of the case is being printed. Note the 'brim' surround that helps it grip the print bed and prevents warping

Now prepare the print bed. We need to ensure the filament will stick to the surface; this is one of the reasons for heating it, to keep the plastic soft. Here, we favour some strips of 'blue' painter's masking tape which you can get from any DIY store. Others use basic masking tape or even combine them with a layer of hairspray or Pritt Glue Stick. Borosilicate glass bases are also very popular and a cheap modification. Check what your printer's manufacturer recommends. If using tape, preheat the bed first or you'll find it moves when it heats up.

> "WE USE OCTOPRINT TO PREPARE AND PRINT A NICE NEW CASE FOR THE RASPBERRY PI"

ON THE LEVEL

The most important part of a successful 3D print is levelling. If your printer is not properly levelled, disaster is waiting for you. We need the nozzle of the extruder to be as perfectly placed above the print bed as possible: 0.2mm when the printer is in its 'Home' position. Too high and the filament won't adhere to the print bed and so will get dragged around, creating a total mess. Too low and you risk the nozzle crashing into the bed, potentially damaging the printer.

We have improved results by preheating the printer before starting to level. Of course, you're now in close proximity to a 200°C 'hot end' and 50–60°C print bed, so be careful and take your time. If unsure, just level at room temperature rather than risk a nasty burn.

Levelling should be done before every print. Instructions vary from printer to printer. We set our i3 printer to 'Home' and then disable the stepper motors so the extruder and bed move freely. Then, we move the extruder to each corner and use a 'feeler gauge' (about £5) which is a small strip of metal 0.2mm thick. You can also use a piece of A4 paper.

If we can get the gauge between the nozzle and the bed and can feel the nozzle pushing against the gauge, things are about right. If not, give the nearest bed screw a quarter-turn and try again. This part takes patience and practice to perfect.

PRINT AND GO

In OctoPrint, click the printer icon next to your GCO file. The printer will adjust temperature if necessary and spring into life. Watch the first layer go down closely. Any issues with levelling, or the nozzle, will become obvious at this point.

3D printing is not a fast game. Our printer took four hours to print each part of the case. Again, OctoPrint comes to the rescue. If you have a Raspberry Pi Camera Module connected, OctoPrint will automatically recognise it. Now, provided it's mounted in a suitable place, you can monitor your print from your laptop or phone. It'll even take a time-lapse video for you.

Now you've got OctoPrint up and running, 3D printing just gets easier and more enjoyable. The fun really begins when you start to create your own models. Software such as Autodesk Fusion 360 and the excellent online Tinkercad (**tinkercad.com**) allow you to design models and download them as STL files. Whatever you can imagine, you can print.

Below When a print is complete, remove the brim and/or supports. You can carefully remove any imperfections with a craft knife. Sand for a smooth finish

10 AMAZING 3D PRINTER PROJECTS

MIND-BENDING CUBE GEARS

A popular printing 'challenge', all the parts print in a single session. They are then assembled to create neat little cubes which, when turned, create the illusion that they are magically held together. **Design by Emmett Lalish.**
magpi.cc/wWEvOM

I AM GROOT!

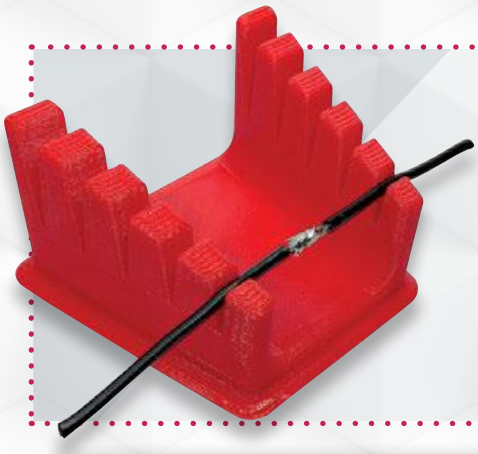
Everyone's favourite tree is now available as a plant pot. This is a great print for modelling enthusiasts, who can sand down and paint the results. It makes a fun gift. **Design by Julia Truchsess.**
magpi.cc/HbEmZU

BRICKS ON DEMAND

A collection of over 5000 LEGO-compatible brick designs and kits. Augment an existing creation or print one from scratch! You can even take the designs and remix them.
printabrick.org

PERFECT MINIMALIST ZERO CASES

The Raspberry Pi Zero is our favourite minimalist computer and it deserves a minimalist case. These cleverly designed cases are quick to print and provide instant protection for our diminutive friend. **Design by Arthur Palmer.**
magpi.cc/XRJTMM



A HELPING HAND

If you've ever needed a third arm when soldering wires together, here's an innovative solution. This straightforward print results in a multi-sized gripper that makes joining wires easy. **Design by Mistertech.**
magpi.cc/fCwQaG

PRINT A RETRO HANDHELD CONSOLE

Why not make your own games console? The good people of Adafruit are no slackers when it comes to 3D printing. Their PiGRRL Zero Raspberry Pi Zero Game Boy is an evolution of the popular PiGRRL series. Add a 2.2-inch TFT 320 × 240 display, battery, and a Raspberry Pi Zero running RetroPie to create a beautiful tiny handheld that can run thousands of retro games. Full instructions are provided with a video tutorial. **Design by Adafruit.**
magpi.cc/rlltYA

IS YOUR PI STARING AT ME?

Fancy some life-blogging? This neat Raspberry Pi Zero case is perfect for capturing time-lapse videos. Just add a Pi, battery, and some cord. It even features a tripod mount and a pocket clip. **Design by Adafruit.**
magpi.cc/Jfiqag

PRINT YOUR OWN ROBOT

The jRobot is an advanced print but a very clever design, available on MyMiniFactory. Using just six parts printed multiple times, you can build up your own tank and power it with a Raspberry Pi. It's a fun way to combine 3D printing and robotics. It'll take patience, as the track link has to be printed 52 times! Once complete, add sensors to create your own custom robot. **Design by Tim Clark.**
magpi.cc/rWBCsC

SWISS ARMY SD CARD

Psst? Need a fresh NOOBs image? How about a Stretch Lite? Or are you after RISC OS? With this Swiss army-style SD card holder, we've got all the images we need right in our pocket. **Design by Dranoweb.**
magpi.cc/lwGiSg

HOW TO PRINT YOUR DRAGON

How cute is this little beastie? Guaranteed 'oohs' and 'ahs' wherever it goes. A clever design incorporates supports that are easily removed. Have fun trying different colours or resize in the slicer to make a family. **Design by Sebastian Popp.**
magpi.cc/eUWQhC

SUBSCRIBE TODAY FROM JUST £4

YOUR OFFICIAL RASPBERRY PI MAGAZINE
The MagPi
The official Raspberry Pi magazine
Issue 69
May 2018
AFFORDABLE 3D PRINTING
Buy your first 3D printer
Use OctoPrint for easy control
All the greatest 3D projects
STREAM YOUR MUSIC
Set up Bluetooth on your Raspberry Pi
NEW GOOGLE AIY KITS
FROM £99
TRANSFORM A RETRO CAM
Kodak Brownie adapted into a modern digicam
GOLD PANNING EVENT
USING A RASPBERRY PI TO PERFORM

SAVE UP TO 35%

£5 FREE! MODMYPI VOUCHER FOR ALL SUBSCRIBERS

MODMYPI

Subscription benefits:

- **FREE!** Delivery to your door
- **EXCLUSIVE!** Raspberry Pi offers and discounts
- **NO OBLIGATION!** Leave any time*

Pricing

Rolling Subscription

- £4 a month
- Quick and easy to set up
- No long-term commitment

* Leave any time applies to Rolling Subscription only

Subscribe for a year:

£55 (UK)

£80 (EU)

£90 (USA)

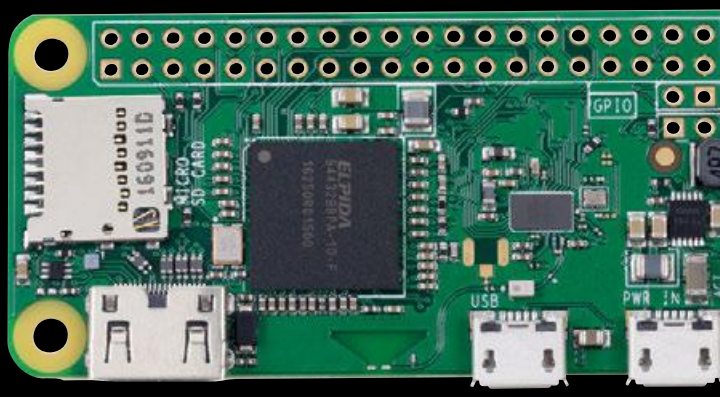
£95 (Rest of World)

magpi.cc/subscribe

JOIN FOR 12 MONTHS AND GET A

PI ZERO W STARTER KIT

WITH YOUR SUBSCRIPTION



Subscribe in print for 12 months today and you'll receive:

- Pi Zero W
- Pi Zero W case with three covers
- USB and HDMI converter cables
- Camera Module connector

SUBSCRIBE ON APP STORES

Available on the App Store

GET IT ON Google Play

FROM £2.29

The software includes clones of Snake, Pong, Flappy Bird, Tetris, Minesweeper, Connect Four, and Conway's Game of Life, as well as slide-shows, animations, board games, and a clock
The table packs an original Raspberry Pi B+, running software written by Andreas and powered by the same 5V supply driving the LED matrix
The system supports up to eight simple controllers and Andreas says they are important for the overall feel of the table

ANDREAS ROTTACH

Andreas is a 24-year-old Master's student of Computer Science, interested in computer graphics/vision, machine learning, electronics, and 3D printing.
magpi.cc/EWRCvg

Quick Facts

- The table measures 70×55 centimetres
- It implements a single LED layout
- The Pi-based software has 7000 lines of code
- The engine supports brightness and player settings
- The software can be updated and recompiled directly

LED-MATRIX TABLE

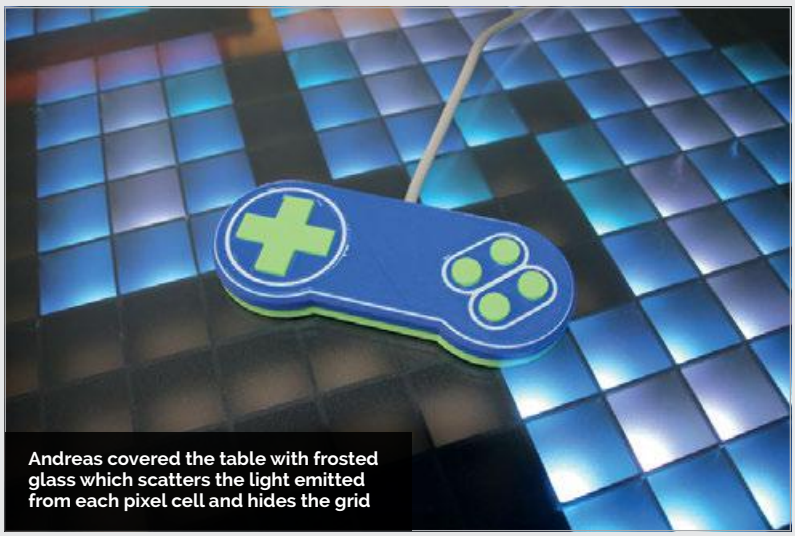
Play retro games and enjoy stunning pixel-based visualisations with Andreas Rottach's multipurpose LED table. **David Crookes** reports

Many coffee-table books have been written about video games, but one thing's for sure: you wouldn't put any of them on this beautiful creation. Neither for that matter would you lay down a cup or a pizza box. After all, you wouldn't want them getting in the way of the retro-inspired action that plays out across its surface.

Created by Andreas Rottach, this project is merely the latest in a line of LED matrix tables to put a Raspberry Pi at its heart. But few look as amazing and well built, helping it to stand out just as much as the gorgeous coloured pixel blocks that make up the display. "Many of these LED tables are used for static animation," says Andreas, "whereas I wanted to focus more on an interactive interface."

Creating classics

Andreas admits he did not plan his project too much. "I just looked for common LED strips and their size and I decided to use a few of these to make it cheap and less wasteful, building the table with that number of LEDs in mind," he



Andreas covered the table with frosted glass which scatters the light emitted from each pixel cell and hides the grid

tells us. The thought, however, was that the table could play a host of classic games. "My first ideas were Pong, Snake, and Tetris" he says. It was then that he enlisted the help of his father.

"My dad focused on building the actual wooden table because I'm not that good at woodwork, but I designed and 3D-printed the matrix grid to keep the pixels sharp and rectangular." The grid, designed to have segments of 3×4 pixels each, was created to overlay the LED matrix and prevent colours from one cell mixing with another. It took more than 70 hours to print, but it was able to work with a strip of 300 LEDs.

> "I used WS2812B LED strips because they are inexpensive"

"I used WS2812B LED strips because they are inexpensive and can be powered with a 5 volt power supply," explains Andreas. "They have a single data wine which is used to set the colour for each individual LED. I cut them into segments of 15 LEDs and chained the data wire from start to end. Each LED processes a bit of the incoming data and transmits the remaining data to the next LED in the chain."

Exerting control

Rather than control each LED individually, Andreas bought a dedicated LED controller. This distributes the RGB colour values on the individual LEDs. The actual image data, however, is generated by software running on the Raspberry Pi. Most of the build time was dedicated to coding this, with Andreas switching from Python to C++ to enhance the performance.

He also used open-source tools such as SDL (**libsdl.org**) for image and audio support, and CMake (**cmake.org**) for building the project software. "I tried to make the software flexible and it is inspired by patterns of game engines for computers. It always runs through the loop: get events, process events, render image."

As time went on, Andreas implemented games of Minesweeper, Conway's Game of Life, and Flappy Bird. He also created a chess-board mode and a clock, and he worked on producing some retro-inspired controllers, although two players can also use a USB keyboard. "It should be very easy to implement any kind of game or visualisation with very little effort," he concludes. "As for the future, I'd love to include a touch interface and maybe have circular outgoing waves when you put down a glass!"

TABLETOP VIDEO-GAMING

>STEP-01

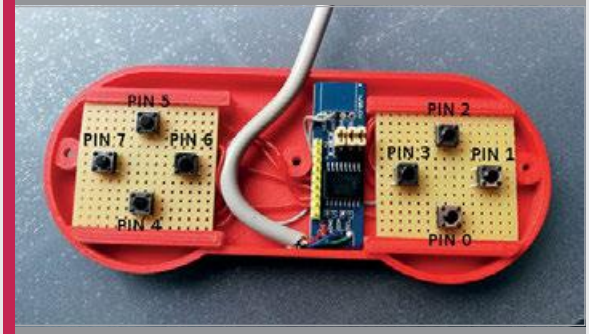
Arranging the LEDs

The WS2812B LED strips have three connections (for 5 V, ground, and data). Laid out in a zigzag pattern, microcontrollers in each LED process incoming data and pass the remaining data to the next LED.

>STEP-02

Laying the grid

This 3D-printed grid overlays the LED matrix. It forms a flat grid with a resolution of 15×20 pixels, preventing colour bleed. An LED controller is connected to one end of the LED matrix.



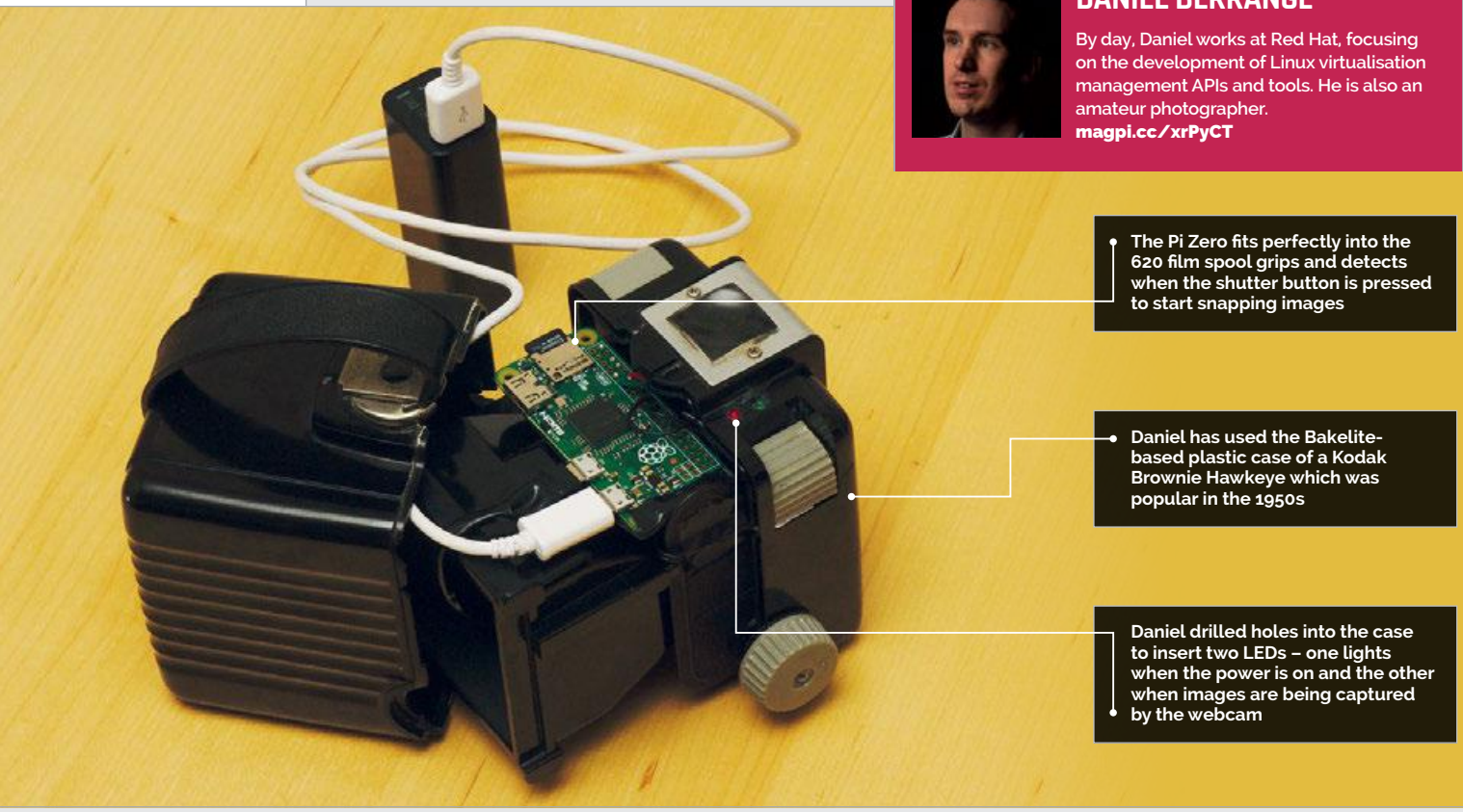
>STEP-03

Exerting control

Andreas based his 3D-printed controller design on SNES gamepads, using simple I²C port extenders plugged directly into the Pi's GPIO pins as the control chip. They allowed him eight buttons.

DANIEL BERRANGÉ

By day, Daniel works at Red Hat, focusing on the development of Linux virtualisation management APIs and tools. He is also an amateur photographer.
magpi.cc/xrPyCT


The Pi Zero fits perfectly into the 620 film spool grips and detects when the shutter button is pressed to start snapping images
Daniel has used the Bakelite-based plastic case of a Kodak Brownie Hawkeye which was popular in the 1950s
Daniel drilled holes into the case to insert two LEDs – one lights when the power is on and the other when images are being captured by the webcam

Quick Facts

- The camera resolution is just 640×480
- But the webcam cost less than £5
- The camera uses a Pi Zero
- The code has been written in C
- The device only needs 185 milliamps to power it

KODAK BROWNIE CAMERA

Daniel Berrangé took an old Kodak Brownie from the 1950s and turned it into a quirky digital camera, as **David Crookes** explains

When Kodak unveiled its Box Brownie in 1900, it did so with the slogan 'You press the button, we do the rest.' The words referred to the ease-of-use of what was the world's first mass-produced camera. But it could equally apply to Daniel Berrangé's philosophy when modifying it for the 21st century. "I wanted to use the Box Brownie's shutter button to trigger image capture, and make it simple to use," he tells us.

Daniel's project grew from a previous effort in which he placed a pinhole webcam inside a ladies' powder compact case (**magpi.cc/FJPZrA**). "The Box Brownie project is essentially a repeat of that design but with a normal lens instead of a pinhole, a real camera case, and improved software to enable a shutter button. Ideally, it would look unchanged from when it was shooting film."

Webcam woes

At first, Daniel looked for a cheap webcam, intending to spend no more than the price of a Pi Zero. This didn't work out too well. "The low-light performance of the webcam was not sufficient to make a pinhole camera so I just decided to make a 'normal' digital camera instead," he reveals.

To that end, he began removing some internal components from the Box Brownie. "With the original lens removed, the task was to position the webcam's electronic light sensor (the CCD) and lens as close to the front of the camera as possible," Daniel explains. "In the end, the CCD was about 15 mm away from the front aperture of the camera, giving a field of view that was approximately the same as the unmodified camera would achieve."

It was then time for him to insert the Raspberry Pi, upon which was

a custom 'init' binary that loads a couple of kernel modules to run the webcam, mount the microSD file system, and launch the application binary. Here, Daniel found he was in luck. "I'd noticed that the size of a 620 film spool (63 mm) was effectively the same as the width of a Raspberry Pi Zero (65 mm), so it could be held in place between the film spool grips," he recalls. "It was almost as if it was designed with this in mind."

Shutter success

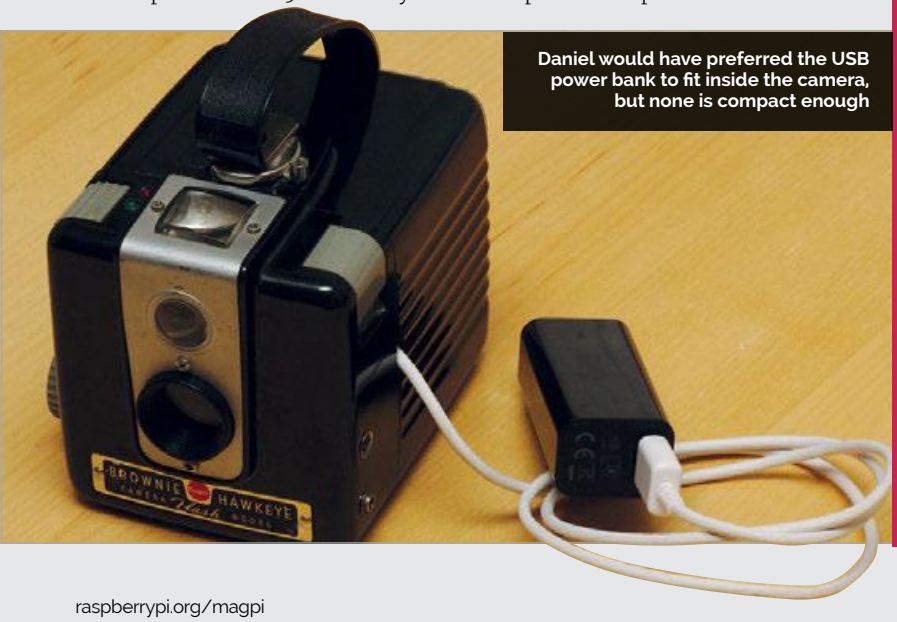
In order to operate the camera, Daniel had to work on the shutter button. "The Box Brownie's shutter button is entirely mechanical, driven by a handful of levers and springs," Daniel explains. "First, the Pi Zero needs to know when the shutter button is pressed and second, the physical shutter has to be open while the webcam is capturing the image. Rather than try to synchronise image capture with the fraction of a second that the physical shutter is open, a bit of electrical tape was used on the shutter mechanism to keep it permanently open."

Above The webcam is inserted into the Kodak Brownie casing, but the camera's aperture doesn't allow light to fully expose the CCD chip, resulting in circular rather than rectangular images

> "Ideally, it would look unchanged from when it was shooting film"

Daniel made use of the Pi Zero's GPIO pins to detect the pressing of the shutter button. It determines if each pin is at 0 or 5 volts. "My thought was that I could set a GPIO pin high to 5 V, and then use the action of the shutter button to short it to ground, and detect this change in level from software."

This initially involved using a pair of bare wires and some conductive paint, although the paint was later replaced by a piece of tinfoil. But with the button pressed, the GPIO pin level goes to zero and the device constantly captures still images until the button is released. All that's left to do is smile and take the perfect snap.



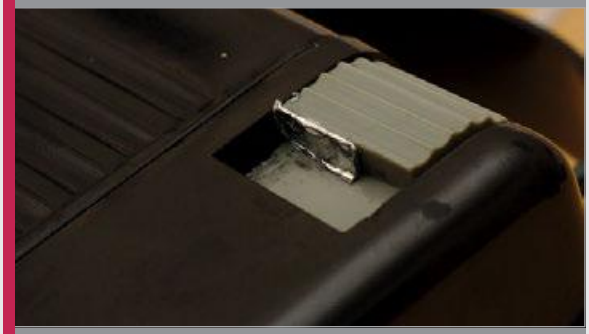
Daniel would have preferred the USB power bank to fit inside the camera, but none is compact enough

START SNAPPING AWAY

>STEP-01

Power up

The camera is powered from a standard USB power bank, so it needs to be plugged in. Upon powering up, it loads the Linux kernel and the custom-written application, illuminating the red LED.



>STEP-02

Press the shutter

The user employs the original viewfinder mirror to centre the subject to be captured, then presses the shutter. The CCD chip has a similar field of view to the unmodified Box Brownie.

>STEP-03

Save the images

A piece of tinfoil glued to the shutter button short-circuits two wires, telling the webcam to capture images. Photos are captured one-per-second and saved to the Pi Zero's microSD card.

ALESSANDRO BENANTI

A prolific maker from Rome, Italy, Alessandro was inspired by the historic Jacquard loom to create computerised machines including a CNC mill, CNC lathe, and robot arms.
magpi.cc/LHKEMJ

CNC EMBROIDERY MACHINE

Quick Facts

- It's based on a 1950s Borletti machine
- It cost around €200 to build
- Maximum speed is 200 stitches per minute
- Three Nema 17 stepper motors are used
- The results are comparable with a modern machine

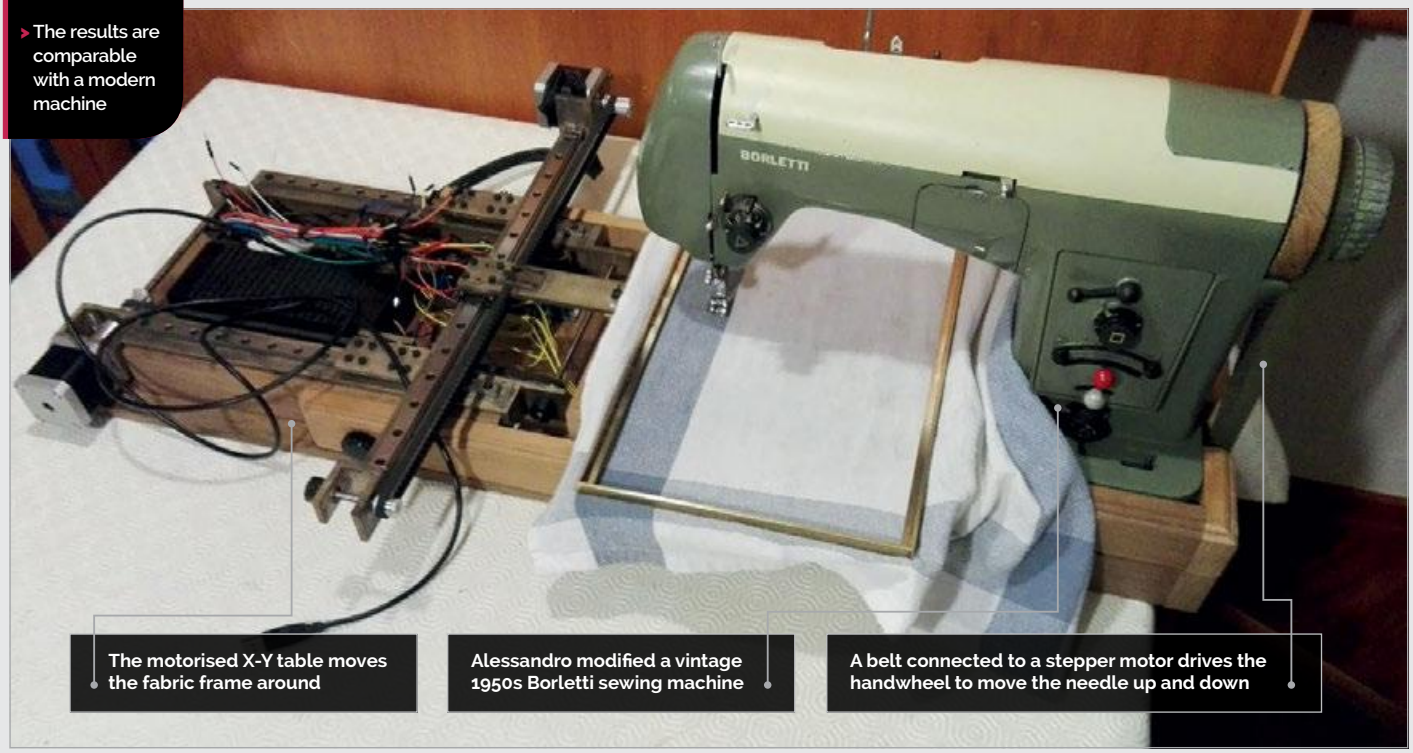
Employing an impressive range of making skills, Alessandro Benanti built his own embroidery machine. **Nicola King** finds out more

Computerised embroidery machines generally cost upwards of £1000, but Italian maker Alessandro Benanti managed to build his own version for around €200 (approximately £175), upcycling a vintage Borletti 1102 sewing machine.

Rome-based Alessandro originally began working on the project back in 2012, but life and other ventures took over, so his sewing machine hack was placed on the back burner for a while. However, he recently found time to complete this intriguing project, which makes use of an original Raspberry Pi Model B.

"I remember the idea came from reading the biography of Alan Turing, when I realised most modern technology evolved from simple machines like the Jacquard loom," recalls Alessandro. Since then he's built other CNC (computer numerical control) machines including a mill and a lathe.

All of his skills were called into play when building the embroidery machine, which involved metal brazing, woodworking, motion control, coding, and designing a graphical user interface.


BORLETTI
The motorised X-Y table moves the fabric frame around
Alessandro modified a vintage 1950s Borletti sewing machine
A belt connected to a stepper motor drives the handwheel to move the needle up and down

He tells us that the 1950s Borletti machine was ideal for the purpose of this project due to its inherent ability to create embroidery designs using sewing foot adapters. "I only cleaned, oiled, and changed the wheel to fit a timing belt. This machine was also an incredible piece of good design, by Marco Zanuso (considered head of industrial design in Italy). Perfect for an attempt."

Sew on and sew forth

Rather than requiring the user to move the fabric around by hand, Alessandro's machine has a motorised system. The fabric is put into a metal frame which is moved along two axes by an X-Y table driven by two Nema 17 stepper motors, controlled by an Arduino Uno connected to a Raspberry Pi.

Above A modified handwheel is driven by a belt connected via a pulley system to a Nema 17 stepper motor, controlled using the z-axis in Grbl on the Arduino

– more than needed by the fabric or design itself."

Alessandro explains that he included an Arduino in the build because "at the beginning this was the only possibility." If building the project from scratch now, he would have avoided this "because with GPIO and LinuxCNC [software to control CNC machines], this could be directly controlled by the Raspberry Pi."

> "I realised most modern technology evolved from simple machines like the Jacquard loom"

A third stepper motor moves the Borletti's handwheel via a belt and pulley system, with an optical switch used to check the needle position in case it goes out of sync.

"In every CNC application, the first movements are to understand the mechanical limits of the machine," says Alessandro. "Every machine has its own limits. Once I found them (acceleration, max speed) the precision is absolute

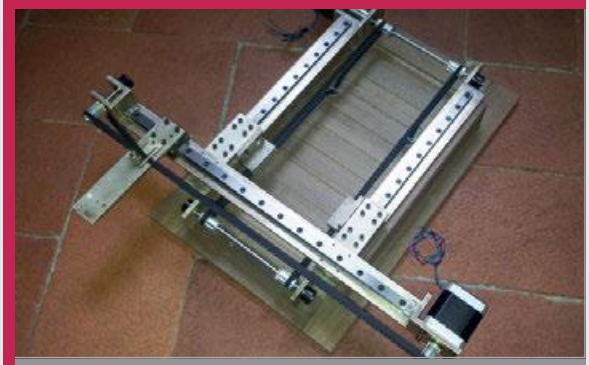
Above The results are impressive, comparing well with those from modern commercial machines

Sew fabulous

Running the CherryPy Python web framework, the Raspberry Pi acts as a web server which can be accessed remotely on any computer. Alessandro designed and programmed a sophisticated graphical user interface for the user to plan their designs using different thread colours.

He tells us there is no noticeable difference in the embroidery results compared to a modern commercial machine. "The only drawback is the speed, which [at 200 stitches per minute] is actually one sixth of the entry-level commercial machines." By changing the wheel motor to a Nema 23, Alessandro hopes to increase the maximum speed.

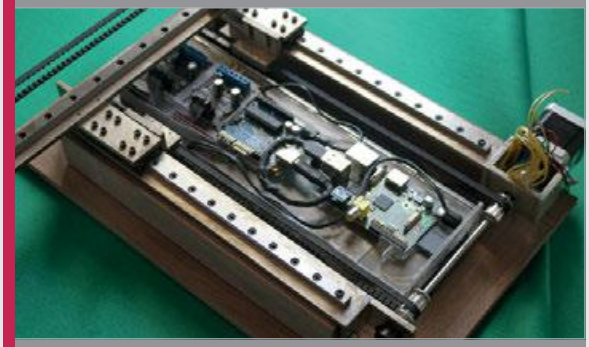
STITCHING IT TOGETHER



>STEP-01

Making the X-Y table

Alessandro designed and built the X-Y table from wooden and brass parts. Nema 17 stepper motors drive two belts to move a brass connector (later attached to the fabric frame) along two axes.



>STEP-02

The electronics

The Raspberry Pi runs a web server and sends data to an Arduino Uno – running Grbl – connected to three PoStep25 boards that drive the motors, including one attached to the machine's handwheel.

>STEP-03

User interface

Connecting to the Raspberry Pi's web server from a remote computer, the user can access a custom-made GUI to create bespoke embroidery designs using a wide range of thread colours.

DARYL BINKS

Based in Japan, Daryl is a Computer System Engineer working in the banking industry. In his free time he enjoys making projects using Raspberry Pi, Arduino, and other electronics.
magpi.cc/SpLiCk

GOLD PANNING TIMER

Seeking a more accurate timing system for the Australian Gold Panning Championships, Daryl Binks made his own electronic timers. **Phil King** digs for nuggets

Quick Facts

- Daryl made 17 timers, including two spares
- The timer software is written in Python
- Panning bays are assigned by RFID tag
- The 5200mAh battery lasts for 14 hours
- Daryl's brother Marcus made the metal stands

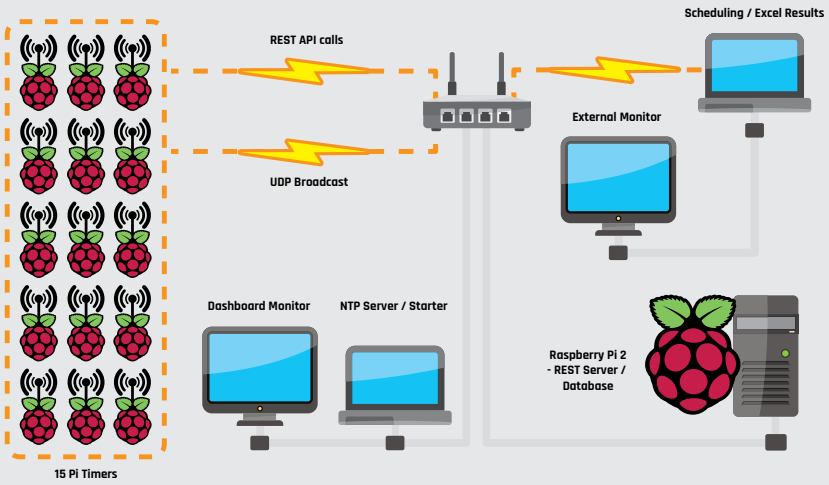
There's a 'gold rush' happening in the tiny township of Blackwood, Victoria. At the sound of the starting horn, competitors at the annual Australian Gold Panning Championships search for gold pieces hidden in buckets of gravel. When a contestant thinks they've found all the nuggets in their bucket (the exact number is unknown to them), their finishing time is recorded, adjusted with a three-minute penalty for each piece missed.

Naturally, an accurate and reliable timing system is vital. "In previous years' competitions, each [competitor] bay was timed by 15 volunteers using stopwatches," says Daryl Binks. "As this is quite inaccurate and prone to discrepancies, a more automated solution was needed, especially since the competitors are getting much faster and the finishing times are closer together."

Tasked by his brother Marcus – Victoria Gold Panning Association president – with finding a cheap solution for an electronic timing system, Daryl spent 18 months creating his own wireless timers. Each timer comprises a Raspberry Pi Zero, LCD, RFID reader, and a

The competitor presses this large stop button when they've finished panning for gold
An LCD shows info such as the heat number and competitor's name
A tag is read by this RFID reader to assign the panning bay
Oct 16 00:26:46
192.168.11.33

Above Daryl's new timers were a big hit with competitors, who had greater confidence in the times recorded



Above The 15 timers are connected wirelessly to two PCs and a RESTful server running on a Raspberry Pi 2

big red stop button: "These are actually emergency stop buttons which are waterproof and when pressed they will lock and stay pressed until released."

After considering 3D-printing the timer case, "Due to the cost restrictions and the requirement to be waterproof, I ended up using a kitchen food container," reveals Daryl. Each timer box is mounted to a backing plate fitted with a metal roof which serves "to help keep muddy water off the display and as a solid mounting for the stop button."

> "Each timer comprises a Pi Zero, LCD, RFID reader, and a big red stop button"

Modular design

One of the design considerations was to make the timers modular so that they could be assigned to any panning bay, allowing easy replacement if needed. This is where the RFID reader comes in. "When the timers first boot up, they are not assigned any specific bay number," says Daryl. "It will wait until we scan the specific bay RFID tag number before it knows which bay it is."

Once all bays are assigned, the starter PC loads the heat information from a spreadsheet and broadcasts it wirelessly. "The timer picks out its own bay number information from the message and displays the heat number and competitor's name on the LCD," explains Daryl.

Hit the button

At the starting horn, the PC broadcasts a 'Start' message to the timers. When a competitor has finished panning all their gravel, they hit the red button. "Once the button is pressed, the timer records the stop time, calculates the delta time between start and stop times, then displays this on the LCD and also sends the information via REST API call to the server [running on a Raspberry Pi 2] which is then written to the database."

Daryl tells us that the timers worked extremely well during the championships, resulting in many positive comments. "The competitors appreciated being in charge of signalling their completion of panning. They went away having confidence in their times. The wireless timers have set a new benchmark for the Australian championships, which the competitors will prefer – if not demand! – for future competitions."

HEY GOLD-TIMER

>STEP-01

Box clever

A waterproof kitchen container holds the Raspberry Pi Zero, along with an RFID tag reader for bay assignment, an LCD screen to show information, and a 5200 mAh USB battery.

>STEP-02

Big red button

The Pi Zero is attached by a cable to a heavy-duty emergency stop button. The latter is screwed to a metal plate which helps protect the timer below from muddy water.

>STEP-03

Out in the field

Placed in a panning bay, the timer receives a start signal. When the button is pressed, the time is calculated, shown on the LCD, and broadcast to a RESTful server running on a Raspberry Pi 2.

ROBERT KATZSCHMANN

A PhD candidate at CSAIL, Robert is the lead author of the *Robotic Fish for Underwater Exploration* report. His previous projects include soft robotic arms and hands.
magpi.cc/kjsBkZ

SOFI

Developed by a team of researchers at MIT, this soft robotic fish swims alongside real ones in the ocean. **Phil King** dives in

Quick Facts

- SoFi can operate at depths of up to 18 m
- A custom PCB is mounted on SoFi's Pi
- An Mbed microcontroller handles the motors
- Ultrasonic echo is filtered by a custom algorithm
- Read the full paper at **magpi.cc/NEdPLE**

In the depths of the South Pacific, a strange new fish is exploring the Rainbow Reef. Flexing its tail from side to side to propel itself serenely along, it captures the underwater scene using a camera – with a fish-eye lens! – mounted in its head, which also contains a Raspberry Pi 2 among other electronics.

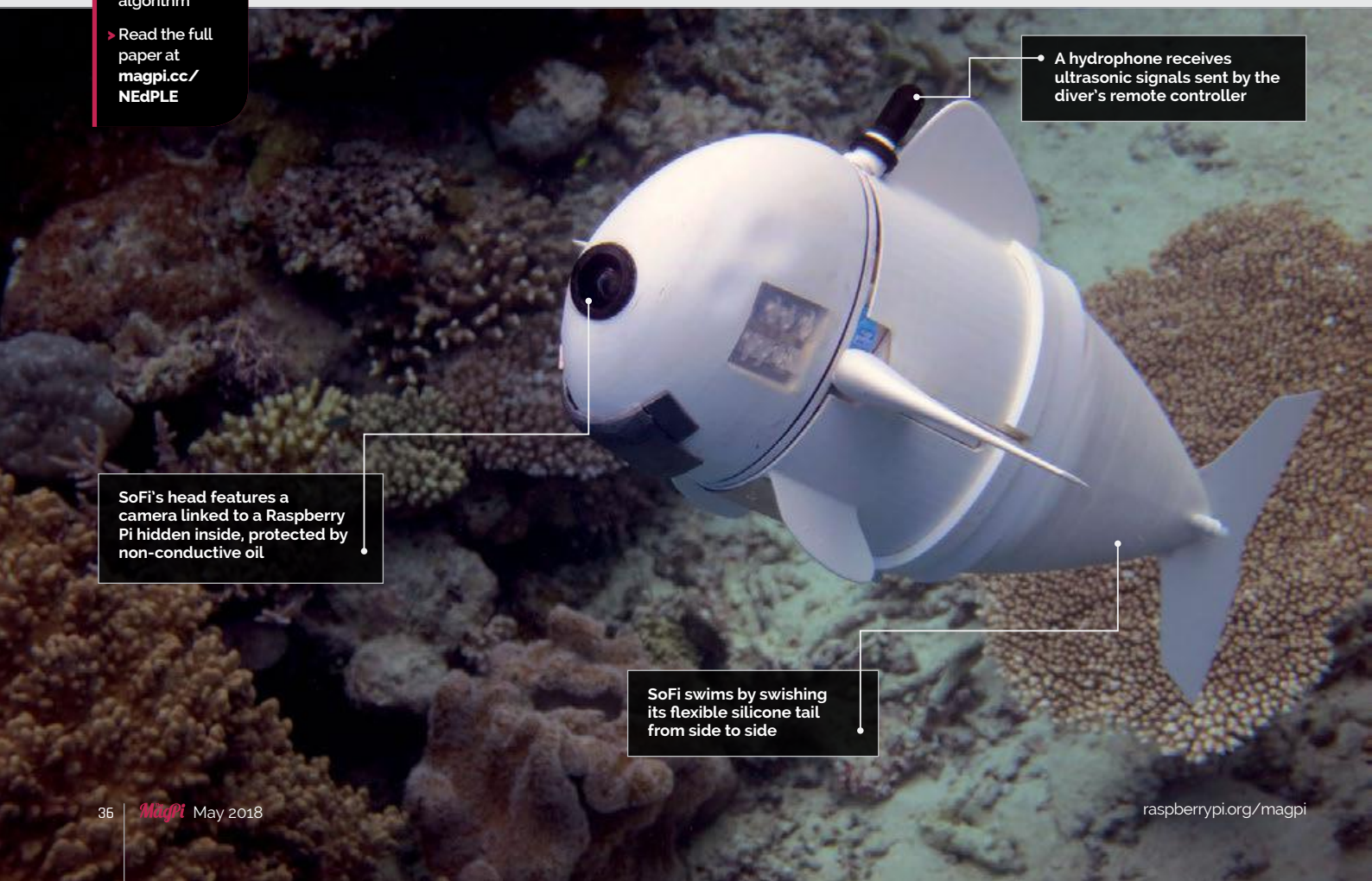
This is SoFi (pronounced 'Sophie'), a soft-bodied robot created by researchers at MIT's Computer Science and Artificial Intelligence Laboratory (CSAIL) to study marine life up close, without disturbing it. That ingenious tail was inspired by the biological system used in tuna fins.

"The fish's motor pumps water into two balloon-like chambers in the tail," explains Robert Katzschmann, lead author of the project. "These work sort of like a pair of pistons in an engine: as one chamber expands, it bends and flexes to one side."

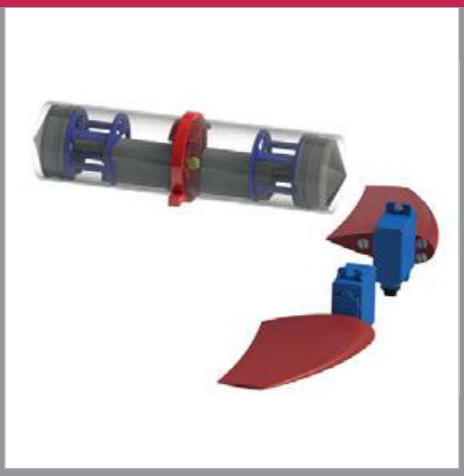
Natural swimmer

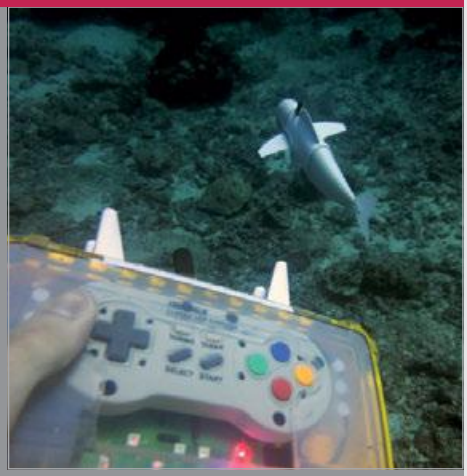
After working on SoFi and its predecessors for more than five years, Robert's team have perfected a naturalistic swimming action. "SoFi can turn, speed up, slow down, and move at different depths, including in strong currents," he reveals. "On average, SoFi swims at a speed of half a body length per second, though we plan to increase this further by improving its pump system and tweaking the design of its body and tail."

The robot has two fins on its side that adjust its pitch for diving up and down, while its overall buoyancy is controlled by an adjustable weight compartment and a chamber that can change its density by compressing and decompressing air.


A hydrophone receives ultrasonic signals sent by the diver's remote controller
SoFi's head features a camera linked to a Raspberry Pi hidden inside, protected by non-conductive oil
SoFi swims by swishing its flexible silicone tail from side to side

SWIM LIKE A FISH





>STEP-01
Swish that tail
A gear motor pumps water alternately into two balloon-like chambers in the silicone elastomer tail, causing it to flex in one direction and then the other to mimic how a real fish moves.

>STEP-02
Dive down
Two side fins are angled to adjust SoFi's pitch to dive down and up. A buoyancy control unit uses compression to adjust the density of air inside it to determine SoFi's overall buoyancy.

>STEP-03
Ultrasonic control
Featuring a second Raspberry Pi connected to a HiFiBerry DAC+, the remote controller sends ultrasonic signals to SoFi. LEDs indicate the currently commanded state of the fish.

"Among some of the challenges we encountered were the strong pressures that our fish had to withstand at deeper depths (down to 18 m) and the boundaries of our acoustic communication range for commanding SoFi remotely," says Robert. A diver uses a waterproof controller, containing another Raspberry Pi, to send commands to SoFi.

"Methods such as WiFi or Bluetooth don't work well underwater, so we chose to use sound instead," explains graduate student Joseph DelPreto. "The remote controller emits ultrasonic acoustic pulses that are too high-pitched for people to hear but that the robot can receive and decode to know how it should behave."

> "SoFi can turn, speed up, slow down, and move at different depths, including in strong currents"

The maximum control range is currently 20 m, but only reliable up to 10 m: "[It] could be higher but we wanted to minimise the disruption to other fish," says Robert.

Following fish

SoFi is also able to navigate autonomously to some degree using its on-board camera. "In the future we will show how SoFi can use its vision to follow other fish," reveals Robert. "By adding pre-recorded maps of the coral reefs onto the Raspberry Pi, we also plan to have the fish self-locate and navigate autonomously through the reefs."

Robert says the team hope to use SoFi to study deep-sea marine life that would be hard to capture otherwise. "The fish can not only gather video, but potentially also other sensor data, as well as taking water samples. We are for example curious to take water samples of the habitats, measure the temperature, and also record the sounds marine animals emit."

SoFi's soft body and naturalistic propulsion system enable it to swim alongside other fish and marine animals without spooking them

101 BLUETOOTH AUDIO

Stream your audio to a Bluetooth speaker

You'll Need

- Bluetooth-enabled Raspberry Pi
- Bluetooth speaker or headphones
- Audio file

RASPBERRY PI MODELS WITH BLUETOOTH

- Raspberry Pi 3
- Raspberry Pi 3B+
- Raspberry Pi Zero W

Bluetooth is a wireless technology standard, and it has been a feature on the Raspberry Pi since the Raspberry Pi 3 and Pi Zero W. You can use it to connect Bluetooth keyboards, mice, and other accessories: like speakers and headphones.

The new Raspberry Pi 3B+ even moves from Bluetooth 4.1 to 4.2, which brings faster and more reliable over-the-air data transmission.

In this tutorial we'll take a look at hooking up a Raspberry Pi to a Bluetooth speaker. This enables you to play music wirelessly from your Raspberry Pi, turning it into an effective media centre.

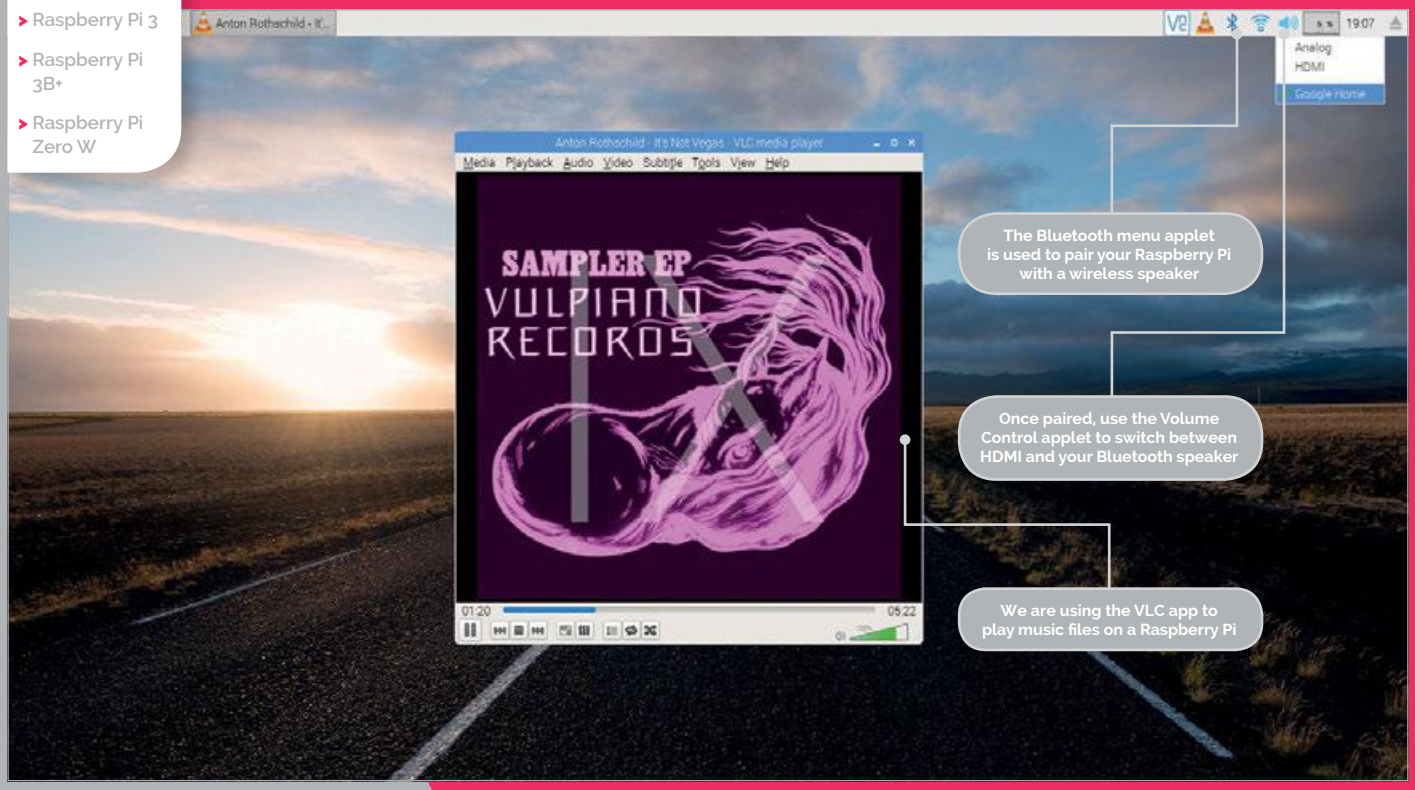
You should be able to follow this Bluetooth audio tutorial with any Bluetooth-enabled Raspberry Pi. And Bluetooth can easily be added to an older Raspberry Pi using a dongle, like this Nano USB to Bluetooth Dongle V2.0 (**magpi.cc/KMBskA**).

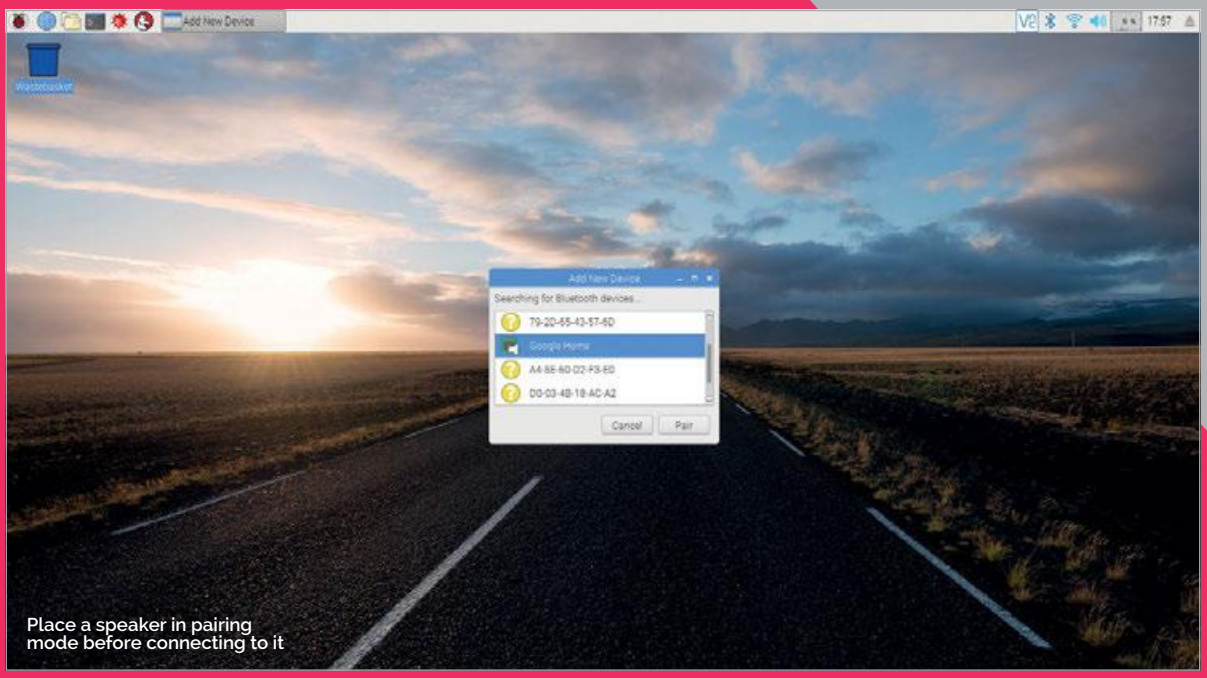
GET SOME MUSIC

First, you'll need some music files. You can stream audio online from a source such as YouTube, and many people have MP3 and other audio files. We downloaded *Sampler EP Vol. 9* by Vulpiano Records Artists from Free Music Archive (**magpi.cc/PAFSkv**) for this tutorial. Save the file to your **Music** folder. Use File Manager to locate the zip file, then right-click it and choose Extract Here.

It is possible to play the music files directly from Terminal using omxplayer. Open a Terminal window and enter **omxplayer** followed by the path to the music file, like this:

```
oxmplayer ~/Music/Anton_Rothschild_-_03_-_
 Its_Not_Vegas.mp3
```


The Bluetooth menu applet is used to pair your Raspberry Pi with a wireless speaker
Once paired, use the Volume Control applet to switch between HDMI and your Bluetooth speaker
We are using the VLC app to play music files on a Raspberry Pi
SAMPLER EP
VULPIANO
RECORDS



Place a speaker in pairing mode before connecting to it

But it's not a great visual option. We'd rather double-click MP3 files in the File Manager using the Desktop interface. For this we need a media player, and we're going to use a popular one called VLC. Enter these commands in Terminal:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install -y vlc
```

During the installation you'll be asked if you want to allow metadata network access. This is used to add cover artwork and track listings to your files. Tick or untick the box and click Continue.

Double-clicking the MP3 files in File Manager will open VLC and play the file.

CONNECT TO BLUETOOTH

OK, so we've got our music and a music player. Now it's time to connect the Raspberry Pi to a Bluetooth speaker.

To do this you'll need to put the speaker in pairing mode and pair the Raspberry Pi to the speaker. How you do this depends on the speaker itself. If your speaker has an interface, you'll probably use the controls to put it into pairing mode. Many smaller speakers or headphones have a button that you hold down to put the device into pairing mode (typically you'll see a flashing light).

We're using a Google Home speaker for this test. For this you simply say "Hey Google... Connect To Bluetooth" (you can do the same with Alexa).

Now click on the Bluetooth icon in the top-right of the screen and choose Add Device.

Look for your speaker in the 'Add new device' window. In most cases the speaker will be identified by an easily recognisable name, such as our Google Home.

However, you may need to locate the device using its MAC address (the six octet hexadecimal code). These are often printed on the device itself – or you can cancel pairing mode, then re-enter it and see which device disappears and reappears.

Hopefully you won't have to resort to that. Locate the speaker in Add Devices and click Pair. After a short while, the window will display Pairing Successful. Click OK.

Your Raspberry Pi is now paired with the speaker, but it won't output audio to it yet. Right-click on the Volume Control applet in the top-right of the screen and choose your speaker from the list (it is below Analog and HDMI).

A pop-up will display 'Connecting to Bluetooth audio device'. Now double-click on of the music files in Files Manager. The VLC app will open and your music will start playing through the Bluetooth speaker.

When you restart the Raspberry Pi, you will need to reconnect to the Bluetooth speaker (this isn't the same as pairing – it is still paired). Click on the Bluetooth icon, choose your speaker, and select Connect from the menu.

You can use the Bluetooth menu to disconnect, if you don't want to use a device for the time-being. Or remove it entirely if you don't want to use it again.

MIKE'S PI BAKERY

MIKE COOK

Veteran magazine author from the old days and writer of the Body Build series. Co-author of *Raspberry Pi for Dummies*, *Raspberry Pi Projects*, and *Raspberry Pi Projects for Dummies*.

magpi.cc/259aT3X

You'll Need

- 2 × Linear Hall effect sensors
- 8 × Ring magnets
- 1 × MPC602 amplifier
- 2 × 10 kΩ trim pots
- 1 × 8-pin DIL IC socket
- 1 × A/D converter, minimum of two channels
- 12 mm diameter dowel
- Various resistors, stripboard, and wood

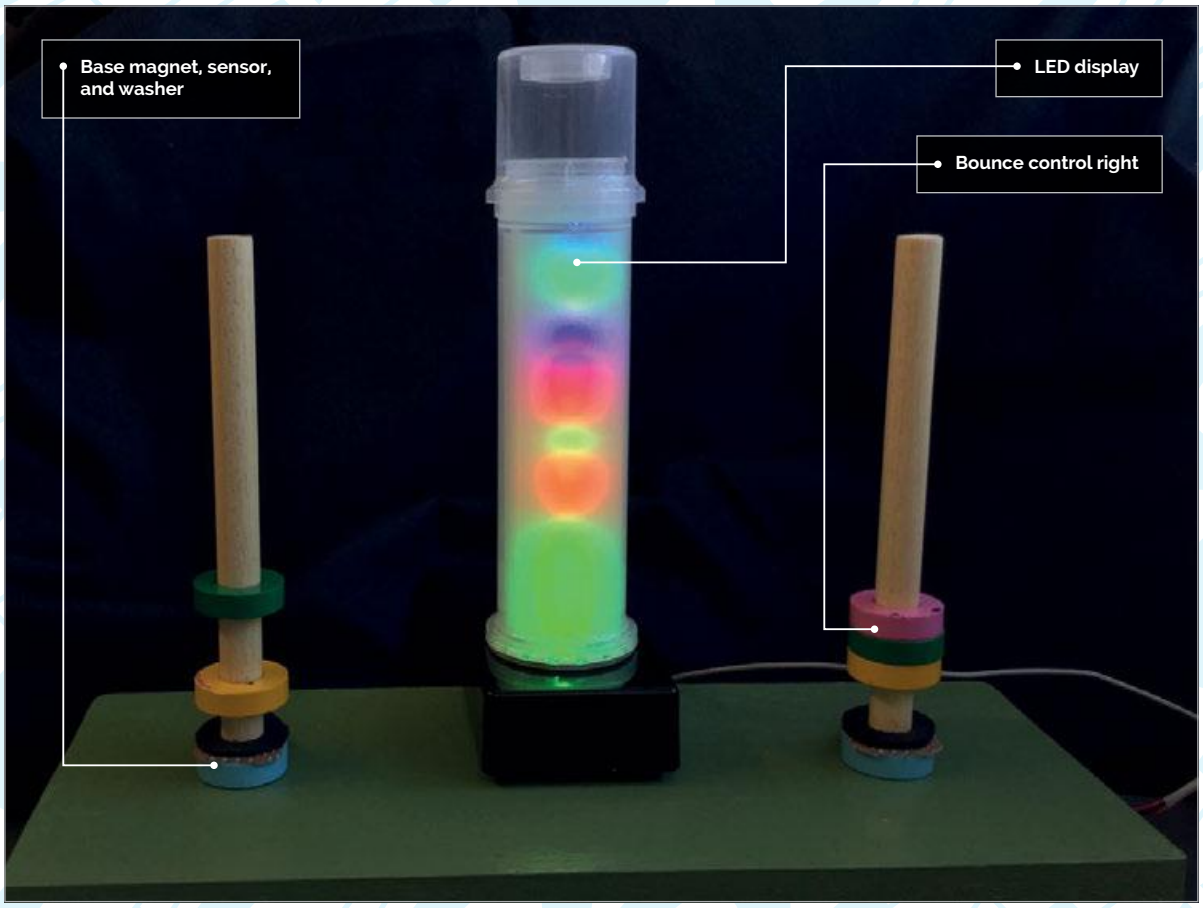
OPTIONAL LED DISPLAY:

- 16 LED WS2812B ring
- 14 LED WS2812B strip
- Empty toothpaste pump dispenser
- Sugru mouldable glue
- Plastic box

CREATE A MAGNETIC BOUNCER

A unique, if crazy, user interface that uses magnets

The magnetic bouncer is a unique computer interface based on the repulsive force of magnets. Magnetic attraction is all very well, but is a bit limited. Bringing two like magnetic poles together so that they repel, however, is fascinating: you can feel the magnetic field as you move them closer together and around one another. A simple way to demonstrate this effect is with ring magnets on a pole: they will sit still with the upper magnet



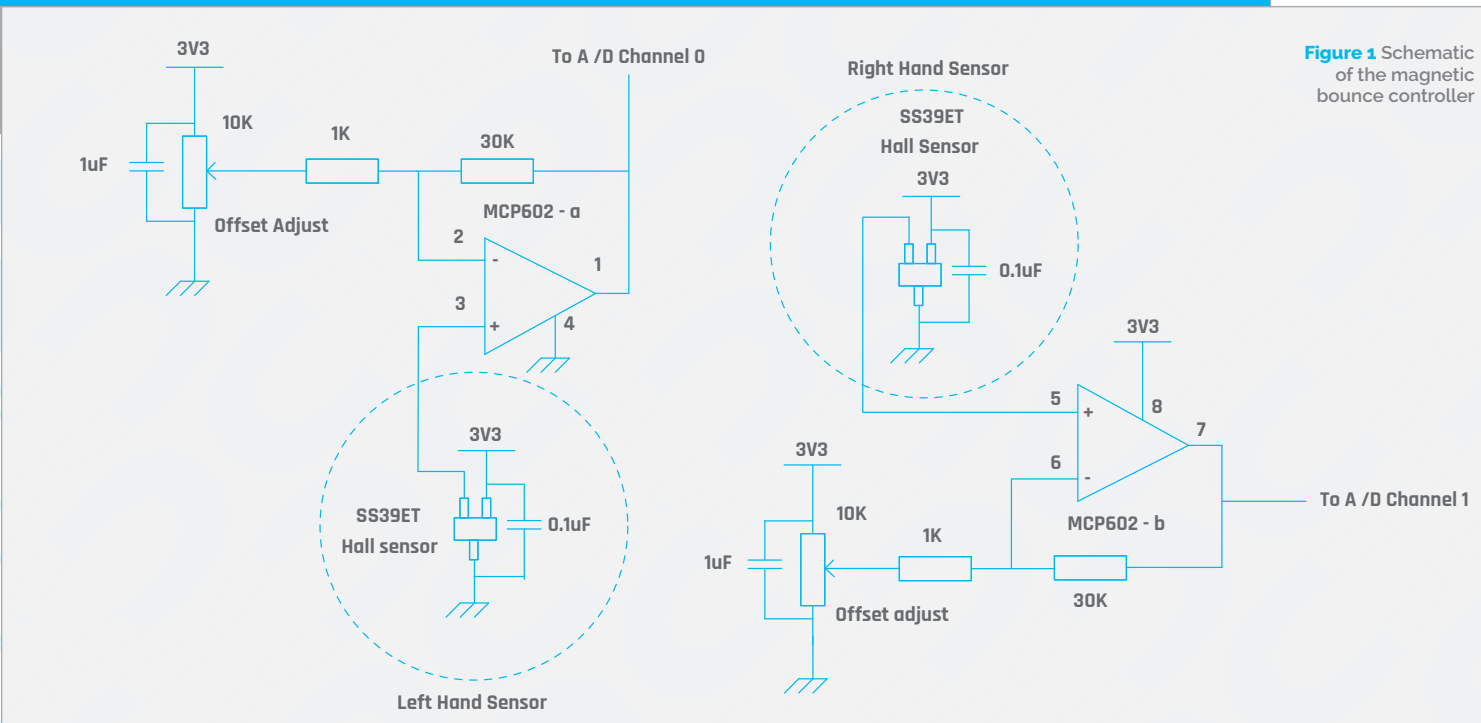


Figure 1 Schematic of the magnetic bounce controller

suspended in mid-air, but disturb them and they bounce around in a most satisfying manner. We wanted to be able to capture this bouncing action and feed it into our computer to act as a controller.

Design

The problem was how to convert this movement into an electrical signal. First off, we tried using a coil between the two magnets, because the bouncing caused a variation in the overall field, and magnetic field variation close to a coil generates an electrical signal. This worked quite well, but only when the magnets were moving, and we felt a direct measure of the magnetic field would be better. Therefore, we used a miniature Hall effect sensor between the two magnets. This meant we had to use an amplifier to remove any offset and increase the voltage from the sensor.

Also, for added flexibility, we made two such systems and mounted them on top of a box. Finally, we thought it would be fun to have this unique controller drive some LEDs in a sort of 'executive toy', so we mounted some LEDs on a small box to sit between the two magnetic bouncers.

The hardware

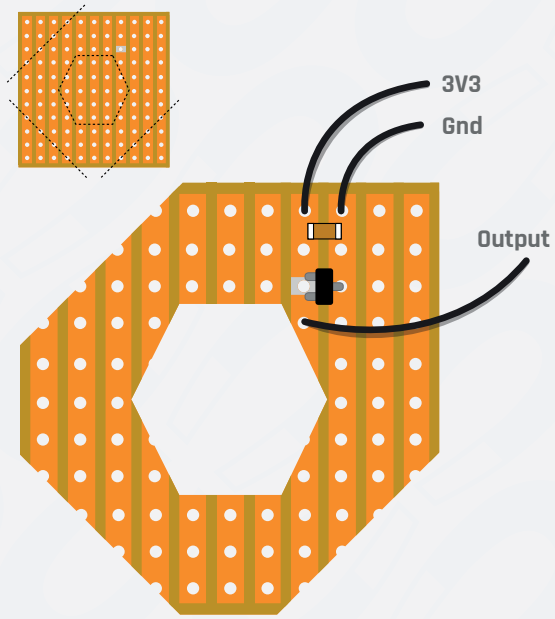
The schematic of the bouncer electronics is shown in **Figure 1**. This shows the two SS39ET linear Hall effect sensors in a dotted circle to indicate they're on a separate board between the two magnets. The rest of the circuit is built on a small piece of stripboard and is located inside the mounting box. This circuit connects to two analogue-to-digital converter channels connected to the Raspberry Pi's GPIO pins. If you don't have one already then you can simply use the one described in the MIDI Drum Sequencer project in *The MagPi* #68.

The LED part of the project is shown in **Figure 2** (overleaf). It consists of a 16-LED ring and a 14-LED strip of the WS2812B type. You need to amplify the 3V3 logic signal out of the Raspberry Pi to be 5 V, to drive this. **Figure 2** shows two ways to do so. The lower circuit uses the 74LS14 inverter IC and was used in our Infinity Mirror project in *The MagPi* #43. The circuit above is a simple FET drive circuit, although you do need

MAKING THE CONTROL INTERFACE BOARD

>STEP-01
Making the sensors

Make two sensor boards, for the SS39ET Hall effect sensor and capacitor. Cut two washers from 1 mm thick foam to prevent the magnets breaking if they bang into each other. Mount this above the sensor board on the dowels.



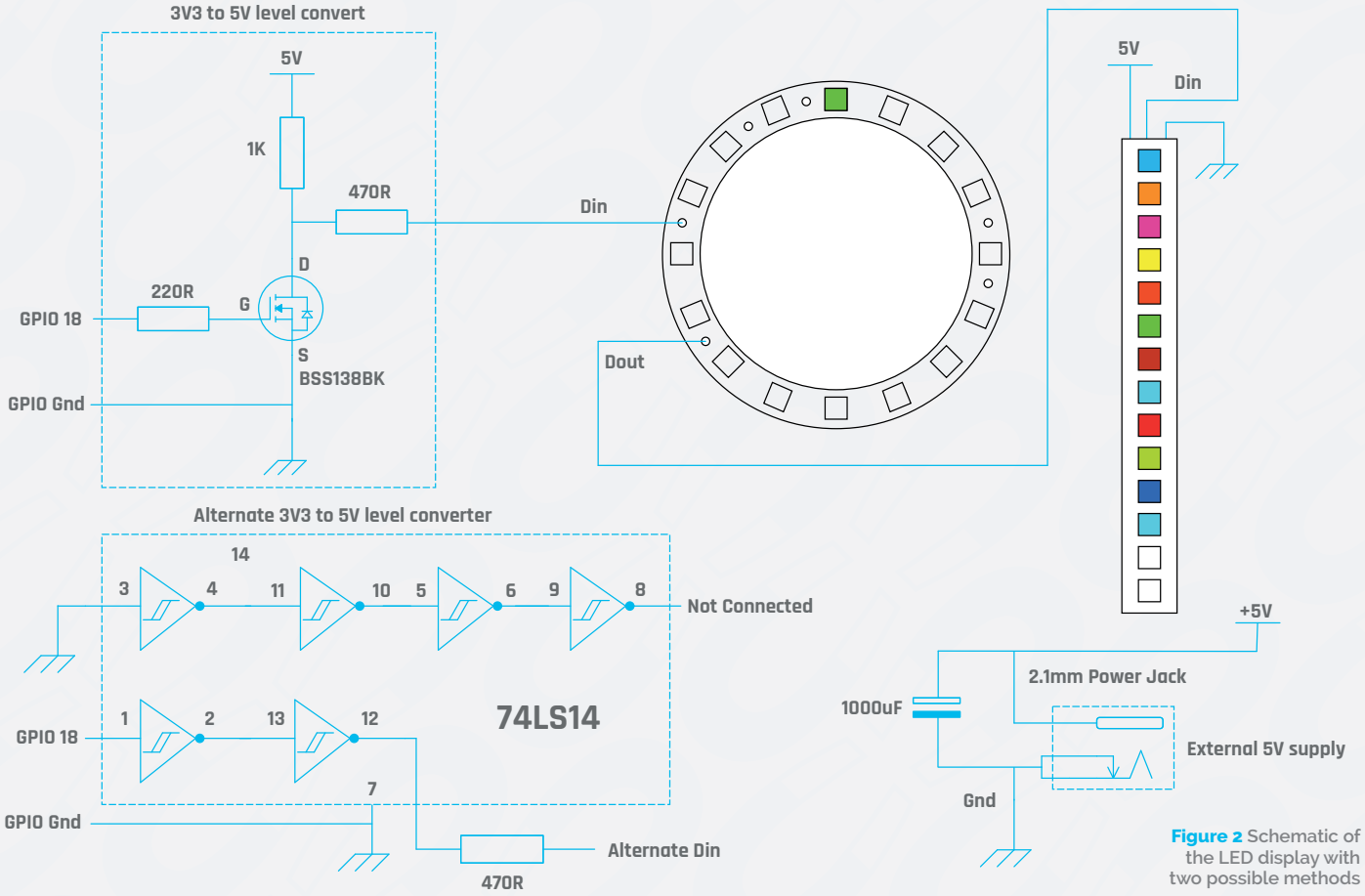
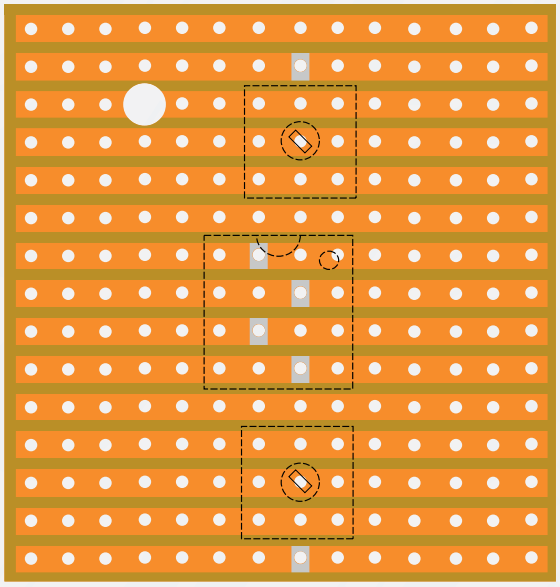


Figure 2 Schematic of the LED display with two possible methods

>STEP-02
Prepare the amplifier stripboard

Take a piece of 14×15-hole stripboard and cut the tracks as marked. Also drill a 3 mm hole in the top-left corner for mounting to the underside of the box.



a FET that will switch off 3V3. Note that due to the number of LEDs, you'll need an external power supply: if all these LEDs were on fully then it would take 1.8 amps, which is too much to take from the Pi's 5 V line. Construction details for both circuits are in the step-by-step guide, with extra photos on the GitHub page.

The software

The first thing we need to do is to look at signals from the sensors and to adjust the preset pots to get the right offset. The code in the **Bounce_test.py** listing will do that and display the results like an oscilloscope. It is written in the Pygame framework and simply reads the two sensors' values and plots the result as a graph. The result is shown in **Figure 3** – note how the top trace shows a magnet configuration of two opposed floating magnets and you can see that it basically consists of two frequencies. The bottom trace has three magnets all stuck together bouncing; due to the greater magnetic field and mass of this configuration, the trace does not decay as quickly as the top trace. The two amplifiers' offset trim pots should be adjusted so that the

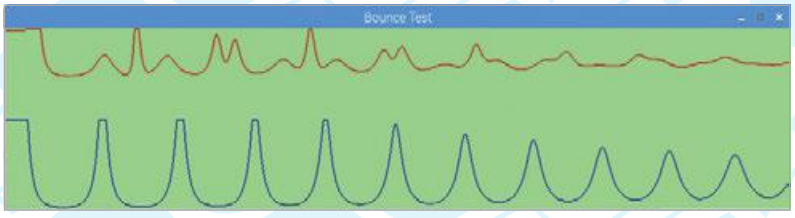


Figure 3 Top trace, left controller; bottom trace, right controller

trace just begins to rise from its lowest point.

Applications

First, we made a drawing application, a sort of Etch A Sketch arrangement. The program can use the two controllers as conventional Cartesian controls – x and y – or as polar co-ordinates: angle and radius. This code can be found on our GitHub page.

Instead of printing this, we thought of using the bounce controller to drive some LEDs in a sort of executive toy style. The idea was to use an LED ring combined with an LED strip placed vertically in the middle of the ring, a sort of totem pole.

One of the controllers determines the colour of the LED and the other controls the position. So a single lit LED rotates around the ring at a speed determined by one controller's movement and each time it passes in front of the totem strip, it adds the current colour of LED to the bottom of the strip and shifts all the others up one. When the totem strip is full, there is a display animation. If at any time you stop bouncing the controllers, the totem slowly loses lit LEDs until all LEDs are off.

The code to do this is shown in the **Neopixel_Bounce.py** listing. You need to have installed the software for Pimoroni's Unicorn HAT to run it. You also need to be in supervisor mode, so use `gksudo idle3` or equivalent.

Taking it further

There are many variations you can make to **Neopixel_Bounce.py**. Maybe the simplest is to change the display animation, or alter the drain animation to run into the ring. You could also build a Pi Zero into the box to make it self-contained. However, the application we are itching to try is to turn this controller into a weird type of theremin, which we shall show you next issue.

>STEP-03
Building the amplifier board

Wire up the circuit as shown in the diagram (right). We used a 5-way DIN socket for the connection between the bounce controller and the A/D converter mounted on the GPIO pins.

>STEP-04
Build the base

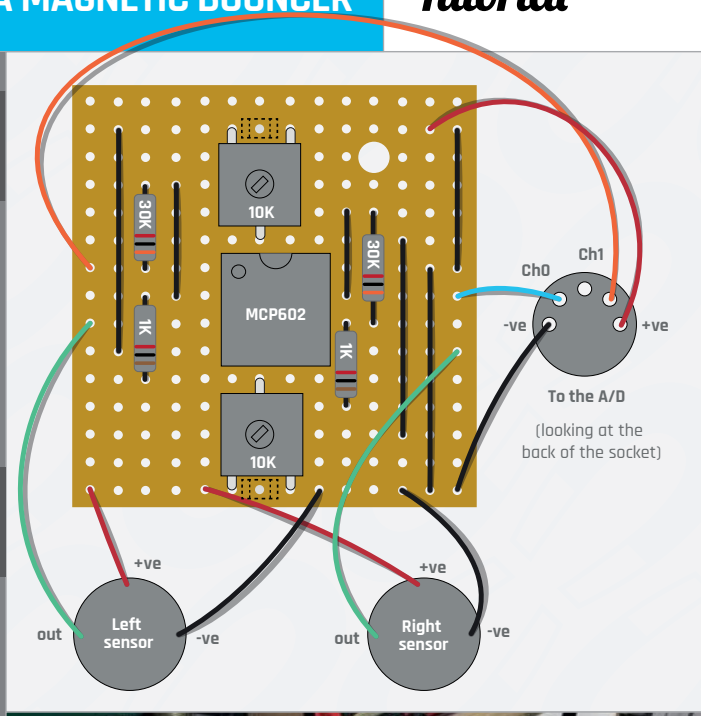
We used 12 mm MDF board to make a box, 300 mm by 120 mm by 40 mm high. Drill two 12 mm holes 190 mm apart and cut the two dowel rods at 165 mm long, insert, and glue. Smooth the dowel with the finest sandpaper you can get and then apply a little beeswax polish to make it glide against the magnets.

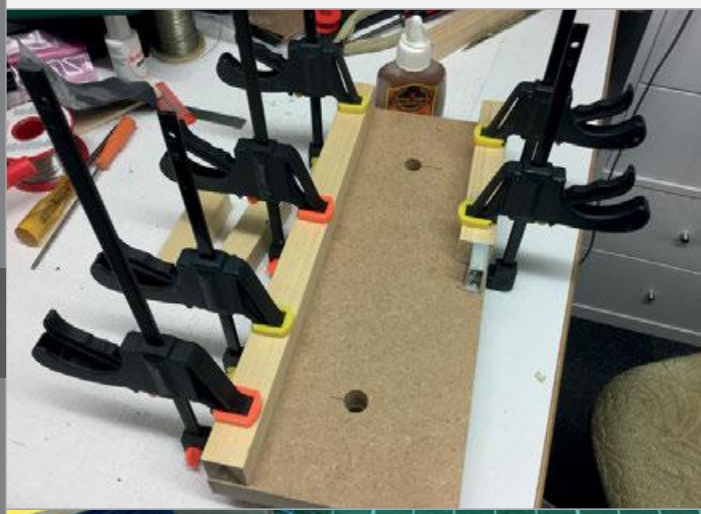
>STEP-05
Building the LED box

We made a ridged strip of LEDs by soldering up 14 of the type that come on their own small PCB with tinned copper wire. We drilled several 2 mm holes in the box to allow the Sugru to get a good grip when mounting the strip vertically in the centre of the circle.

>STEP-06
Building the LED box

A Colgate toothpaste dispenser was disassembled so we just got the tube and top. The base exactly fits the LED ring. Slip some greaseproof paper or polyester sheet on the inside of the dispenser to act as a light diffuser. Glue this onto the LED ring with a PVA-type glue that dries transparent. Fix the LED box to the main box with self-adhesive Velcro strips.





Neopixel_Bounce.py

```
#Neopixel Bounce - controlling LEDs with the bounce interface
#**** must start IDLE3 with "gksudo idle3" *****#

import time , spidev
from neopixel import Adafruit_NeoPixel

# LED strip configuration:
LED_COUNT      = 30      # Number of LED pixels.
LED_PIN        = 18      # GPIO pin connected to the pixels (must support PWM!).
LED_FREQ_HZ    = 800000  # LED signal frequency in hertz (usually 800KHz)
LED_DMA        = 5       # DMA channel to use for generating signal (try 5)
LED_BRIGHTNESS = 140     # Set to 0 for darkest and 255 for brightest
LED_CHANNEL    = 0       # PWM channel
LED_INVERT     = True    # True if using an inverting interface

ws2812 = Adafruit_NeoPixel(LED_COUNT, LED_PIN, LED_FREQ_HZ, LED_DMA, LED_INVERT, LED_BRIGHTNESS, LED_CHANNEL)
ws2812.begin()

length = LED_COUNT
circleLength = 16
totemLength = 14
launchPoint = 11 # LED opposite totem
inBuf = [0, 0] ; lastBuf = [0, 0] ; difBuf = [0, 0]
totBuf = [ (0,i,0) for i in range(0,totemLength) ]

def main():
   print("Neopixel Bounce - Cnt C to stop")
   initHardware() ; curCol =(0,120,0)
   wipe() # clear all LEDs
   place=0 ; totCount = 0 ; stoped = True
   while True:
      wipeC(0,circleLength,(0,0,0)) # blank circle LEDs
      if not stoped :
         set_led(place,curCol[0],curCol[1],curCol[2]) # current colour
      if place == launchPoint and difBuf[1] > 4: # right place and moving
         for i in range(totemLength-1,0,-1): # add to totem
            t = totBuf[i-1]
            totBuf[i] = t
         totBuf[0]= curCol
         transToTot(length) # transfer all
         totCount +=1
         if totCount > totemLength :
            runEffects()
            totCount = 0
            wipeC(circleLength,circleLength+totemLength,(0,0,0))
            for i in range(0,totemLength):
               totBuf[i] = (0,0,0)
      ws2812.show()
      readSensor()
      if stoped and difBuf[0] > 4:
         stoped = False
      s = mapV(difBuf[1],0,500,0.1,0.002)
      curCol = setCol()
      time.sleep(abs(s))
      if difBuf[1] > 4 :
         place += 1
         stoped = False
         stopedTime = time.time()
         if place >= circleLength:
            place=0
      else: # slowly decay
         if time.time() - stopedTime > 2.0:
            stoped = True
            if totCount > -1 :
               if totCount >= totemLength :
                  totCount -=1
               totBuf[totCount] = (0,0,0)
               stopedTime = time.time()
               totCount -= 1
               if totCount < 0:
                  totCount = 0
               transToTot(length) # transfer all
               ws2812.show()

def runEffects(): # display when totem fills up
   for j in range(0,4): # ascending LEDs
      wipeC(circleLength,circleLength+totemLength,(0,0,0))
      ws2812.show()
      time.sleep(0.3)
      for i in range(0,14):
        transToTot(circleLength+i+1)
        ws2812.show()
        time.sleep(0.1)
   for i in range(0,10): # flash totem LEDs
      wipeC(circleLength,circleLength+totemLength,(0,0,0))
      ws2812.show()
      time.sleep(0.2)
      transToTot(length)
      ws2812.show()
      time.sleep(0.2)

def transToTot(size): # transfer totem buffer to LEDs
   j=0
   for i in range(circleLength,size):
      set_led(i,totBuf[j][0],totBuf[j][1],totBuf[j][2])
      j+=1

def setCol(): # HSV colour space with S = V = 1
   h = abs(inBuf[0])
   while(h > 255):
      h -= 255
   if h < 85:
      return (int(h * 3), int(255 - h * 3), 0)
   elif h < 170:
      h -= 85
      return (int(255 - h * 3), 0, int(h * 3))
   else:
      h -= 170
      return (0, int(h * 3), int(255 - h * 3))

def wipeC(s, e,col): # wipe with a colour
```

```
    for i in range(s,e):
       set_led(i,col[0],col[1],col[2])

def mapV(x, in_min, in_max, out_min, out_max):
  return (x - in_min) * (out_max - out_min) / (in_max - in_min) + out_min;

def wipe(): # everything off
    for i in range(0,length):
       set_led(i,0,0,0) # black
    ws2812.show()

def set_led(i, r, g, b):
    if i < LED_COUNT:
        ws2812.setPixelColorRGB(i, r, g, b)

def readSensor():
   lastBuf[0] = inBuf[0] ; lastBuf[1] = inBuf[1]
   for i in range(0,2):
      adc = spi.xfer2([1,(8+i)<<4,0]) # request channel
      inBuf[i] = (adc[1] & 3)<<8 | adc[2] # join two bytes together
   difBuf[0] = abs(inBuf[0] - lastBuf[0]) # work out changes
   difBuf[1] = abs(inBuf[1] - lastBuf[1])

def initHardware():
   global spi,lastX,lastY,ch0Low,ch1Low
   spi = spidev.SpiDev()
   spi.open(0,0)
   spi.max_speed_hz=1000000

# Main program logic:
if __name__ == '__main__':
   try:
     main()
   except: # clear up the LEDs
      wipe()
      ws2812.show()
```

Bounce_test.py

```
import pygame, os, time, random
import spidev

pygame.init()
os.environ['SDL_VIDEO_WINDOW_POS'] = 'center'
pygame.display.set_caption("Bounce Test")
pygame.event.set_allowed(None)
pygame.event.set_allowed([pygame.KEYDOWN,pygame.QUIT])
screenWidth = 1000 ; screenHight = 230
screen = pygame.display.set_mode([screenWidth,screenHight],0,32)
textHeight= 20
font = pygame.font.Font(None, textHeight)
backCol = (150,255,150) # background colour

inBuf = [ 0, 0]

def main():
    n=0
    loadResource()
    while(1):
        time.sleep(0.001)
        checkForEvent()
        readSensor()
        display(n)
        n +=1
        if n > screenWidth:
          n=0
          lastX = -1; lastY = 0
          pygame.draw.rect(screen,backCol,(0,0,screenWidth,screenHight+2),0)

def display(n):
    global lastX,lastY
    col = (180,64,0)
    y0 = ch0Low - inBuf[0]//9
    y1 = ch1Low - inBuf[1]//9
    if n != 0:
       pygame.draw.line(screen,col,(lastX ,lastY[0] ),(n ,y0 ),2)
       pygame.draw.line(screen,(0,64,180),(lastX ,lastY[1] ), (n ,y1 ),2)
    lastX = n
    lastY[0] = y0 ; lastY[1] = y1
    pygame.display.update()

def readSensor():
   for i in range(0,2):
      adc = spi.xfer2([1,(8+i)<<4,0]) # request channel
      inBuf[i] = (adc[1] & 3)<<8 | adc[2] # join two bytes together

def loadResource():
   global spi,lastX,lastY,ch0Low,ch1Low
   spi = spidev.SpiDev()
   spi.open(0,0)
   spi.max_speed_hz=1000000
   pygame.draw.rect(screen,backCol,(0,0,screenWidth,screenHight),0)
   lastX = -1 ; lastY = [0,0]
   ch0Low = screenHight/2 -2
   ch1Low = screenHight -2

def terminate(): # close down the program
    pygame.quit() # close pygame
    os._exit(1)

def checkForEvent(): # see if we need to quit
    event = pygame.event.poll()
    if event.type == pygame.QUIT :
        terminate()
    if event.type == pygame.KEYDOWN :
       if event.key == pygame.K_ESCAPE :
          terminate()
       if event.key == pygame.K_d : # screen dump
          os.system("scrot")

# Main program logic:
if __name__ == '__main__':
    main()
```

Language

>PYTHON 3

DOWNLOAD:
magpi.cc/1NqJjmV

PROJECT VIDEOS

Check out Mike's Bakery videos at:
magpi.cc/DsjbZK

PETER JUETT

Peter is a veteran computer programmer, has worked across industries and continents, and has a passion for Raspberry Pi and Python.
twitter.com/peterjhk15

CREATE A TOUCHSCREEN CONTROL AND DATA SYSTEM

You'll Need

- 7-inch touchscreen and stand
- BMP180 – temperature / pressure (optional)
- GY30 – light (optional)
- DHT12 – temperature / humidity (optional)
- PIR – movement (optional)

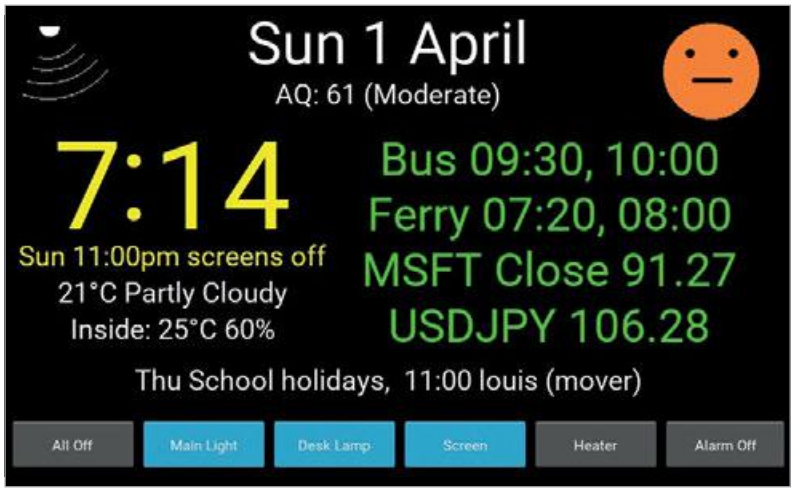
Build and customise a control and data system to wake you, remind you, inspire you, and control your surroundings

From humble beginnings with the clock and local transport schedule, this system grew with needs and wants and includes the weather, temperature, humidity, stock prices, air quality, and even a family calendar!

There are several of these great touchscreens around our home that show everyday information, control the surroundings, and take care of some repetitive tasks.

Of course, there are apps on our phones that do some of this, but there is nothing quite like having data at a glance and control at your fingertips; and all completely adaptable and easily extendable.

The system is composed of lightweight message-enabled software modules which run independently on your main touchscreen Raspberry Pi, or other Pi boards around the home. The modules communicate with each other by exchanging Mosquitto messages and responding accordingly. In fact, each of the screen labels shows data from its associated message, and each of the buttons and events transmits a message or set of messages. We can configure them to our heart's desire. Here's some of what we have set up:



Left The screen in all its glory! Various information, plus motion detection indicator and a smiley that changes mood depending on the air quality!

DISPLAYS

- Time and date
- Air quality (WHO)
- Temperature and humidity (inside and out)
- The weather
- Stock prices and exchange rates
- Google calendar
- Detected motion

MANUAL CONTROL

- Lights over the Hue system
- Mains sockets using TP-Link Smart Plugs

AUTOMATIC CONTROL

- Day, time
- Temperature, humidity
- Light level
- Motion

It also speaks, albeit with inspirational quotes and the time mainly, but also the occasional polite message.

Voice actors (from **fiverr.com**) recorded numbers and greetings and the inspirational quotes, which you can find in the GitHub repository: **magpi.cc/eLVgEr**.

The system is open and extendable and, perhaps a bit of a cliché, but it's only really limited by your imagination.

Critically, for privacy, none of your data or activities is sent anywhere on the internet, with just some simple calls to APIs for weather, air quality, etc. It's completely under your control. The open-source nature of the system gives you complete transparency – no black boxes here.

Nor is there any machine learning or artificial intelligence, nor voice control. However, with the flexibility here, this is all possible should you wish to extend it to include these features.

Install the system

To start setting up the system, you need to install the Kivy Pie build (**magpi.cc/OHAIyS**) – this is an image of Raspbian Jessie with Kivy already installed and ready to go. Kivy is a cross-platform library that allows you to program the touchscreen to your heart's content!

Next, install the Mosquitto MQTT libraries for the communications:

```
sudo wget http://repo.mosquitto.org/debian/
mosquitto-jessie.list
sudo apt-get install mosquitto
sudo apt-get install mosquitto-clients -y
sudo pip install --upgrade pip
sudo pip install paho-mqtt
```

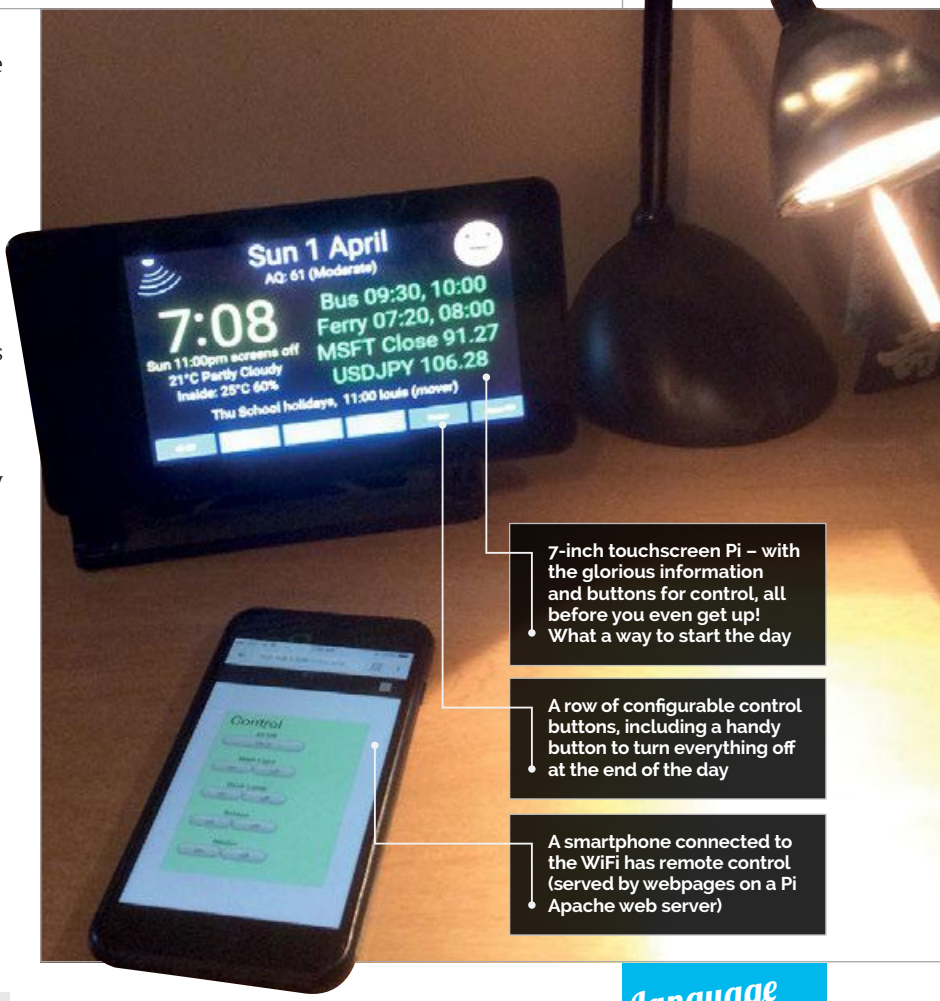
Install SQLite3 database libraries and set up the database to store settings, message profiles, and 'macros':

```
sudo apt-get install sqlite3
```

You'll then need to install the Apache web server and PHP. Apache serves up the webpages for remote control and configuration. PHP is used to drive interactive webpages which store information to the database and transmit commands using Mosquitto:

```
sudo apt-get install apache2 -y
sudo apt-get install libapache2-mod-php5 -y
sudo apt-get install php5-sqlite
```

Install the source code modules from the GitHub repo (the Python modules, required libraries, webpages, and subfolders). You can find more details in the **README.md** file on GitHub.


Sun 1 April
7:08
Bus 09:30, 10:00
Ferry 07:20, 08:00
MSFT Close 91.27
USDJPY 106.28
7-inch touchscreen Pi – with the glorious information and buttons for control, all before you even get up! What a way to start the day
A row of configurable control buttons, including a handy button to turn everything off at the end of the day
A smartphone connected to the WiFi has remote control (served by webpages on a Pi Apache web server)

How to customise it

Now we can customise the screens to our needs. We do this from the webpages, selecting which messages to associate with which screen labels, which text for the buttons, and whether they are toggle (two-state buttons – for lights and such) or simple buttons (single-state for alarm off and such).

Once we have done this, we can then assign messages to the buttons and events. Make sure not to forget to go to the configuration screen and set up your own custom settings for emails and IP address for our Mosquitto communications along with some other goodies.

In a browser, go to the 'Display and Buttons': **http://<Raspberry Pi IP address>/GlanceWeb/displayandbuttons.php**

For each of the labels, select the message that you want to display. For the buttons, enter the text you want to appear on each button and the type of button (single/two-state) and save.

Then go to the 'Events and Actions': **http://<Raspberry Pi address>/GlanceWeb/eventsandactions.php**

For each event, enter a name, then which days it should be active (we can enter weekdays, weekends, everyday, Mondays, etc.), and also a time.

Now, for the actions, assign the list of messages to trigger. This is how we set up the button's scheduled events and what to trigger on high/low sensor events.

Language
>PYTHON

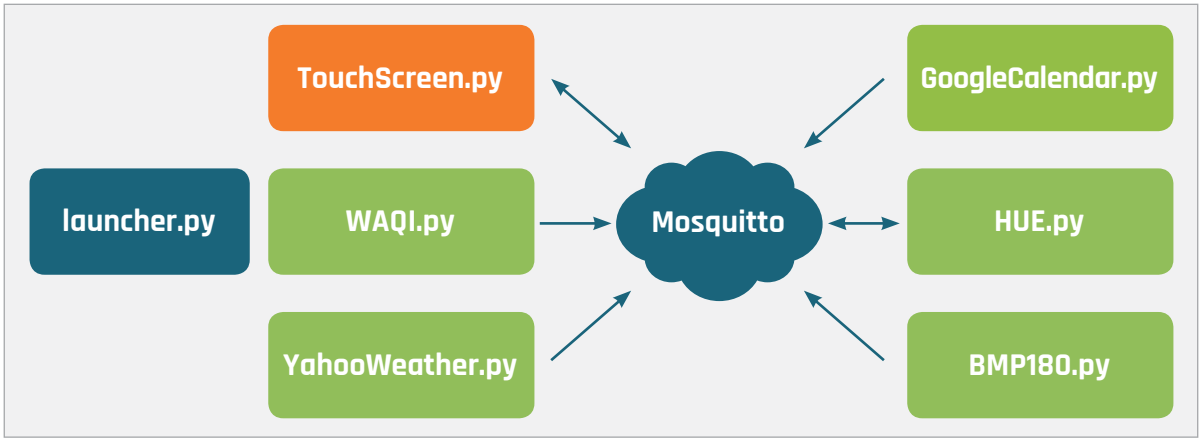
DOWNLOAD:
magpi.cc/eLVgEr

STATIC IP ADDRESS

Use static IP address for your Raspberry Pi so it won't change when you restart the wireless router.

TROUBLE-SHOOTING

Check out the log files. In the **/home/sysop/Glance/logs** directory, use `tail -f launcher.log` for watching the log in real-time.



Right The modules can run standalone, or be bundled and launched together with the launcher.py script – handy for organising a distributed network of these

Next, go to 'Configuration': **http://<Raspberry Pi address>/GlanceWeb/config.php**

Under Mosquitto communications, set the Broker address to match your Raspberry Pi's IP address, and make sure the port matches your setup (in reality we can probably leave this port as set).

Resist the temptation to change the name, for the moment at least (this is actually used as the host name in the Mosquitto messages).

Next, we should check that the **launcher.py** Python script suits our requirements. On the Raspberry Pi, navigate to **/home/sysop/Glance** and open the **launcher.py** file:

```
sudo nano launcher.py
```

The **launcher.py** script simply spawns a thread for each of the modules listed in the 'bedroomtouch' section.

This list of modules for each Raspberry Pi is a great way to keep things organised (especially with a distributed system of several Pi devices).

How to run it

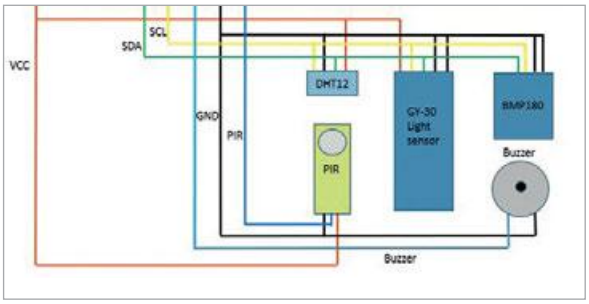
We can run the touchscreen and sensor modules using **launcher.py**. Go to **/home/sysop/Glance** and run:

```
sudo python launcher.py &
```

TRIED TO RUN AND NOTHING HAPPENED!

If you are not running the system headless, be sure to type in the sysop password from the Kivy Pie FAQ: **magpi.cc/tsYXZu.**

The **TouchScreen.py** script works by using `run_program` at the bottom of the script, which creates the main application class, `MyApp()`.



Above Evolved through experience – a custom sensor and buzzer board

MyApp is responsible for creating the screen class, **MainScreen**, and also holds application-level variables and contains the Mosquitto code for communications. **MainScreen** sets up the labels, buttons and icons, and callback methods for updating the screen.

Data and control with HelloWorld.py

As mentioned previously, the system is scalable, flexible, and extendable with the modules following a standard template. Each module has a specific purpose, whether gathering data from the internet, the sensors, or controlling something cool! Using **HelloWorld.py** as an example, you can then easily adapt it for other data sources, actions, and interfacing with other systems.

First, the object is initialised, and creates the Mosquitto client, assigns the callback methods, and connects to the host (using values stored in the database).

```
self.start_mosquitto()

self.mos_client.on_message = self.on_message

self.mos_client.connect(self._db.get_
value("mosbrokeraddress"), int(self._db.get_
value("mosbrokerport")), 60)
```

The 'on-message' callback method we assigned earlier will receive messages and is responsible for processing them. In this example, we check the host and the message to be sure it is the message that we are waiting for, and then just store the received value (extracted from the third part of the message).

```
def on_message(self, mosclient, userdata,
msg):

if messageparts[0] == self._db.get_
value("name"):
   if messageparts[1] == "SetHelloWorld":
     self.set_data(messageparts[2])
```

We use the 'Publish' loop in this example to simply to send the data onto the Mosquitto network, periodically. Other modules will periodically poll the internet for data and share on the Mosquitto network, or trigger a send on change of GPIO state, etc.

```
while(1):
   self.mos_client.publish(self._db.get_
value("mostopic"), self._db.get_value("name")
+ "/HelloWorld/" + self.get_data())
   time.sleep(SLEEP_TIME)
```

Remote control

As we saw earlier, we can remotely control the system from a smartphone! This works thanks to the Apache web server, which serves up the database-driven webpages, displaying the buttons that we configured earlier.

When we click one of these buttons, Mosquitto messages are sent from the PHP behind the control webpage according to the command sequence we configured. These messages are then intercepted by the relevant Python module which, in turn, triggers the code to be executed perform the action; e.g. switch the I/O on the Raspberry Pi.

All in the database

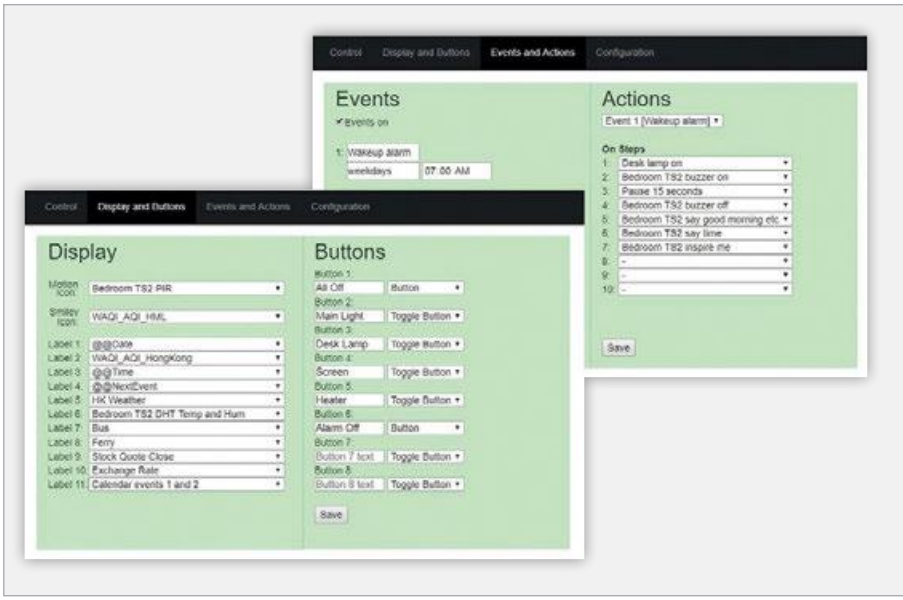
The database contains the settings and message definitions for the system. Most of the settings are configurable from the webpages; e.g. Mosquitto broker IP address.

The database also holds configurations for the screen layout, messages associated with screen labels and buttons, message sequences (i.e. the 'macros'), and event schedules

There are two tables, 'settings' and 'messages', intentionally kept generic and simple:

```
CREATE TABLE settings(
   id INTEGER PRIMARY KEY AUTOINCREMENT ,
   setting           TEXT    NOT NULL,
   value             TEXT    NOT NULL
);

sqlite> .schema messages
CREATE TABLE messages(
   id INTEGER PRIMARY KEY AUTOINCREMENT ,
   host TEXT    NOT NULL,
   name TEXT    NOT NULL,
   value TEXT    NULL,
   description TEXT NULL,
   display TEXT    NULL, -- '1' - this
message is suitable for screen display
   action TEXT    NULL - '1' - this message
is for triggering an action.
);
```



Above Configuring the 'what to show', 'what to do', and 'when to do them'

One point to note is that all the messages used in the system are listed in the messages table. The host must match the host name of the Raspberry Pi to which they are associated.

The messages table also provides a more human-readable description. This is used in the message selection from the webpages.

From the Python code, each of the modules loads and reads the database:

```
self._db = DB.DB()
self._db.load_settings()
```

Then refer to the settings with something like **`self._db.get_value("mosbrokeraddress")`** and refer to a message like **`self.db.get_message("lightsensor")`**.

The wonders of I²C

It would be remiss of us not to include what you can do in terms of sensors. We put together a collection of some of our favourite sensors, a buzzer, a PIR motion sensor, and created a PCB to connect them all, for convenience mainly.

The I²C bus is neat, just requiring two connections for communications (plus power) and bundling these sensors up with motion detection and the buzzer. Using something like a PCF8574 I²C to I/O chip, you can interface the buzzer and PIR to the I²C, and expand the I/O at the same time. Alternatively, you can connect them directly to the GPIO using the the RPIIO.py module.

We hope you find this system as useful and fun as we did and join the party to continue to evolve it, should you wish. These touchscreens are great little units and with the Raspberry Pi, internet, sensors, and data, the possibilities are endless.

CONTROL FROM THE INTERNET

You can host the webpages externally by using an online MQTT broker (**magpi.cc/sZJIkF**). Remember to stay secure!

3DAMI

This tutorial can be found on the Raspberry Pi resources website and was written by b3d101, Peter Kemp, Tom Haines, and Monique Dewanchand.
magpi.cc/mfhwVV

BUILD A 3D-PRINTABLE ROCKET

Learn Blender basics by building a rocket in the 3D-modelling software, then turn that into a 3D printer file

You'll Need

- A PC capable of running Blender
- Blender – **magpi.cc/wxxwdG**
- 3D printer – see page 16

If you've followed our earlier tutorial on how to build a 3D printer, you may be looking for things you can print with it. You may also be wondering how to make your own 3D models to print.

Blender is free and open-source 3D-modelling software, which can easily be used to make your own models for 3D printing. It can be a bit tricky to get your head around, though, so we're going to show you how to get started with Blender by creating a rocket in it.

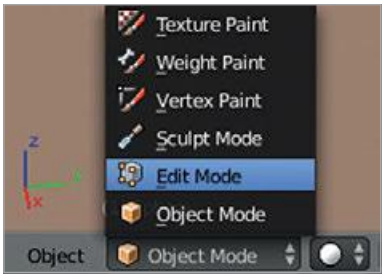
Unfortunately, Blender doesn't currently work on the Raspberry Pi, so for now you'll need another computer.

Base of the rocket

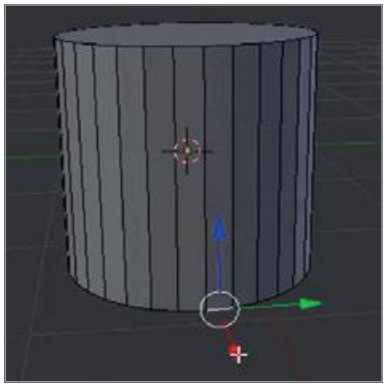
Open Blender and exit the splash screen. Start by selecting the cube, then press **X** to delete it. Now go to the Toolbox (on the left) and click on the Create tab. Add a cylinder.

We'll be creating the rocket from this cylinder. The cylinder will be the bottom of the rocket, and we'll add to the top of it to build up the rocket.

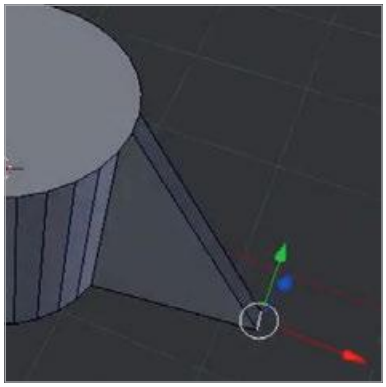
Switch to Edit Mode using the menu at the bottom.



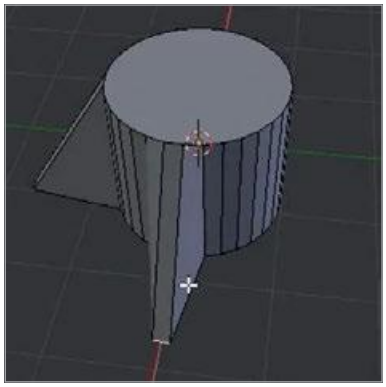
Let's create the fins that stabilise the rocket as it flies. Click on Edge Select in the lower menu: it looks like a cube with a line running down one of the edges. Then right-click an edge at the base of the cylinder to select it.

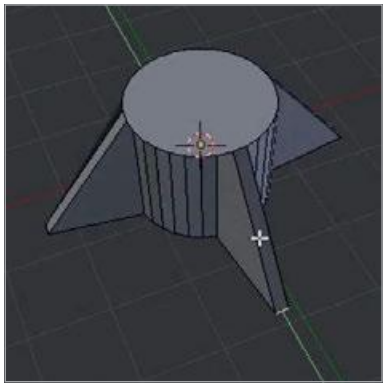


Click and drag the edge out using the coloured arrows to create a fin.



Now repeat this three more times to create four fins – make them symmetrical!



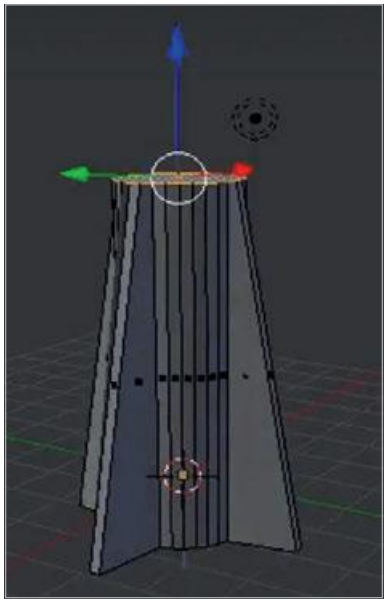


Now you have the base of your rocket!

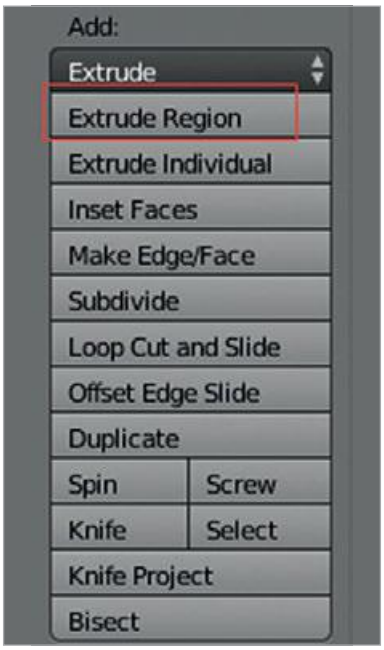
Rocket body

Now let's create the body of the rocket. Switch to the Face Select tool: it's next to the Edge Select tool and shows a cube with one of the faces highlighted in yellowy-orange. Next, right-click to select the top face of the cylinder.

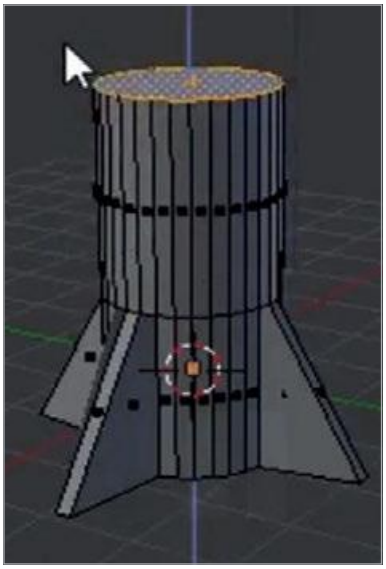
If you pull it up, it looks a bit strange. Everything will move up, and that's not what you want.



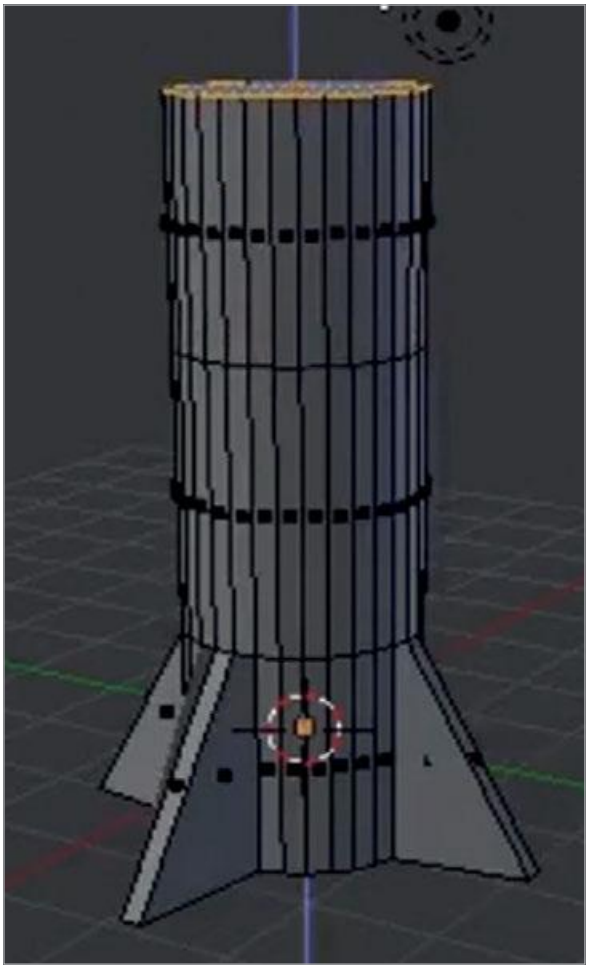
Press **CTRL+Z** to undo any changes you made. Instead of pulling it up like this, we will extrude this face. Go to the Toolbox, select the Tools tab, then select Extrude Region.



Pull the section up to create a taller rocket, then click when you are happy.



Extrude the same face once more to form the basis of the nose cone. Either select Extrude Region again from the menu, or if you prefer, use the shortcut key **E**.



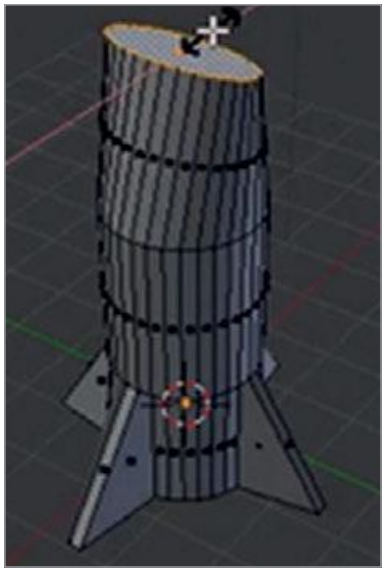
3D AMI

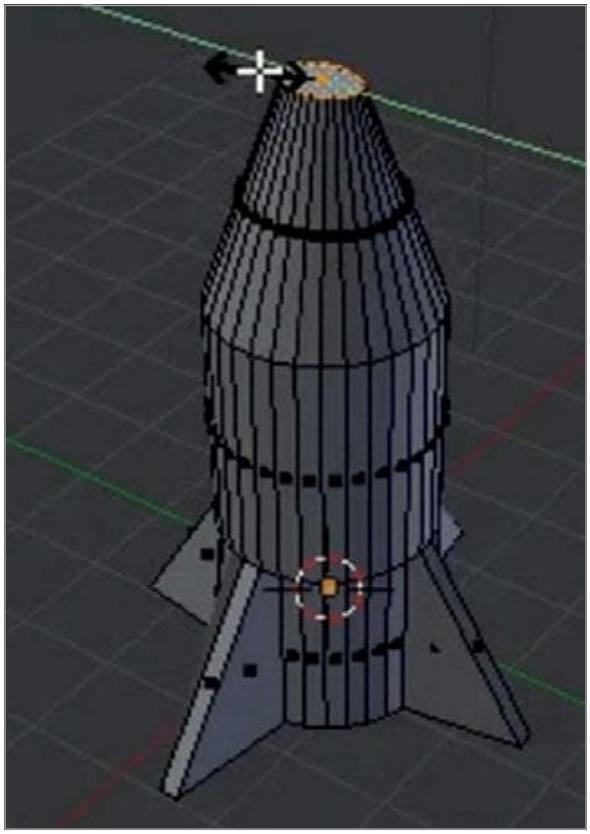
These tutorials have been adapted from the version on the Raspberry Pi projects website – **rpf.io/projects** – but were originally created by 3D Ami under a Creative Commons Attribution 4.0 International License (**magpi.cc/BsYUbU**).

Make a nose cone

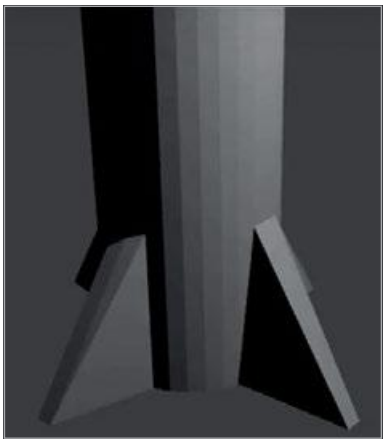
The nose cone should come to a point. To do this you can use the resize tool. Do this by selecting the Cube End tool: it looks like a small blue square with a diagonal line drawn off the bottom-left corner.

Resize the face using the red and green handles, turning it into more of a nose cone shape.

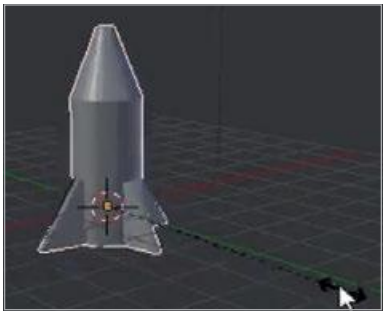




Render the image by pressing **F12** (**FN+F12** if you're using a Mac) and see what the rocket looks like.



It might look a bit big, but you can resize the whole rocket; first press **ESC** to exit the render view. Switch back to Object Mode using the menu at the bottom, then press the **S** key to resize, then shrink the rocket. Click when you are happy with the size.



Render again by pressing **F12** (or **FN+F12**) to confirm the rocket is of the correct size. Press **ESC**.

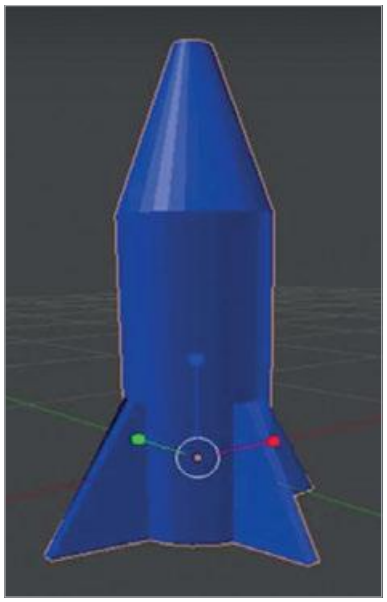
Colour the rocket

First we will give the rocket a basic blue colour. Select the rocket and then select the Material tab on the right-hand menu panel.

Add a new material by clicking New.

Call this new material 'blue'.

Select a blue colour from the Diffuse menu palette to colour your rocket blue.



Add stripes

Make sure the rocket is selected, then select Edit Mode from the lower menu. If you prefer, you can use the **TAB** key as a shortcut.

Next, click on the Face Select tool from the lower menu again.

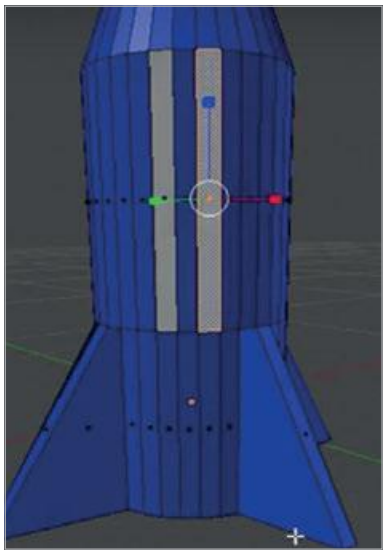
Now we will add a new material for the stripes. Go to the Material tab on the right-hand menu panel, and add another new material by clicking the + icon.

Click on New and call the material 'white'.

Select a white colour from the Diffuse menu like with the blue material. Now you can add this colour to the faces of the rocket.

Right-click to select a face on the rocket that you want to colour white. Next, click on the white material, then click on the Assign button to colour the face white.

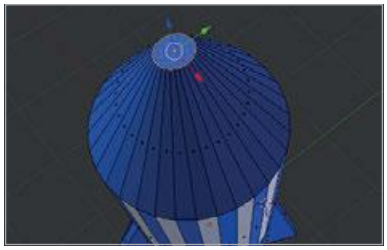
Now select the next face to be coloured and give it the white colour by repeating the previous steps on every other face. You can select multiple faces at once by selecting a face and then holding **SHIFT** while selecting the other faces.



Render your rocket to see what it looks like.

Give it a red nose

Let's colour the nose red. Right-click to select the face of the nose.



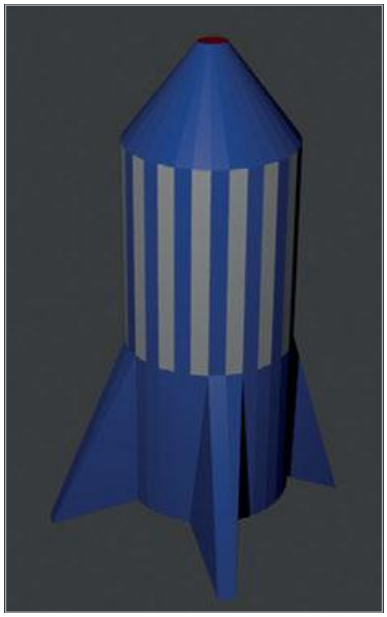
Add a new material called 'red' and select a red colour from the Diffuse menu. Assign the red material to the nose, and then render to see your blue rocket with white stripes and a red nose.

Making it 3D-printable

For it to work in our 3D printer project from this issue (see page 16), you'll need to export your Blender file as an STL. Save the rocket file, and then select it.

Go to File > Export > STL. Tick the Scene Unit box. Give the file a name, and click Export STL.

You may need to experiment with the size of the 3D model and the print parameters when printing this and other Blender files, but with a bit of practice you'll be able to print out just about anything you desire.



MORE 3D-MODELLING PROJECTS

Want to learn more about Blender? Here are some more tutorials to try...

TREE OF CUBES magpi.cc/xIYnFG

In this tutorial, you'll learn how to resize objects by creating some trees in Blender. These trees are a bit op-heavy so might not be good for 3D printing as they are. Can you figure out how to modify the tree to make it more easy to print? You can try support struts or even breaking it down into smaller components to fit together.

SNOWMAN magpi.cc/bESvMB

Work with spheres to make a snowman! Spheres are interesting items to practise with for 3D printing, as the gradual layering allows them to expand from the base. For 3D printing purposes, though, you may want to edit the final version so that it has a flatter base so it can start printing properly.

BLOCK HOUSE magpi.cc/FZPHnI

Take a single block and turn it into a house. There's more about extruding and manipulating an object in this tutorial, and it's also an interesting project for 3D-printing. Would you try and print it in one piece, or break it down and attach them once printed?

BRIAN BEUKEN

Very old game programmer now teaching very young game programmers a lot of bad habits at Breda University of Applied Science in Breda NL.
scratchpadgames.net

CODING GAMES ON THE RASPBERRY PI IN C/C++ PART 05

Time to get our graphics to move and shake with code, and start making a playable game

You'll Need

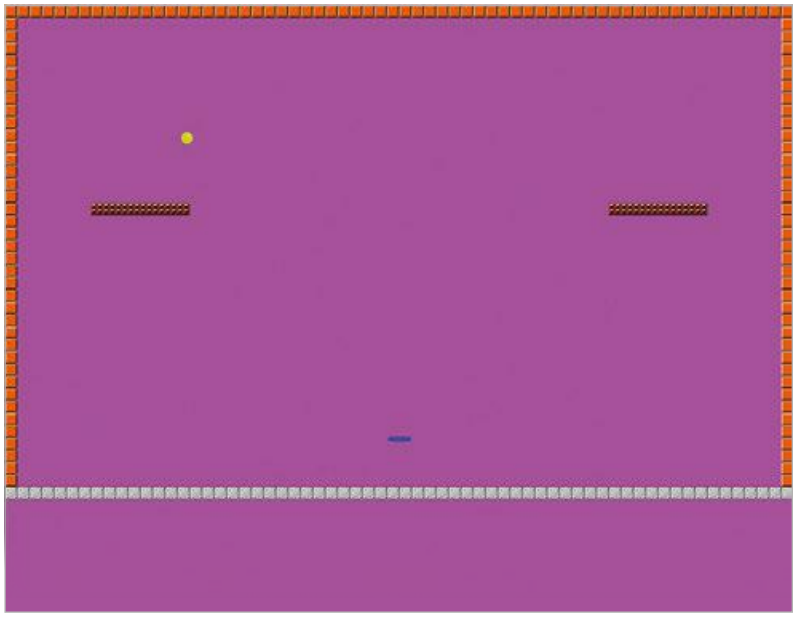
- Code::Blocks
  `sudo apt-get install codeblocks`

Finally we have some working systems. They're not good, but they will do for now while we focus on doing something more interesting. We'll improve the systems later when needed, but let's first learn how to move and shake.

We have the ability to draw a playfield. Now let's start to interact with it and bring in the tiles' distant cousin, the sprite.

We have graphics now?

Our draw system is currently slow, but we've got enough to work with. So we can consider adding some moving things that we can control on screen.



Below
Our bat and ball are at the ready

We already worked out how to move our triangles and squares, so moving a square with a different graphic on it is just as straightforward.

It's important to note one thing, though: our screen is not our playfield; rather it's a visual representation of our playfield. Which in this case is held in a grid or 2D array. We're actually going to interact with that grid and draw the results on screen.

With that in mind, our ability to use pixel positions for placement becomes way more useful to us, since we can directly map a pixel co-ordinate to a cell co-ordinate in our grid, simply by dividing the 'screen' position by the size of the tiles.

Let's add sprites

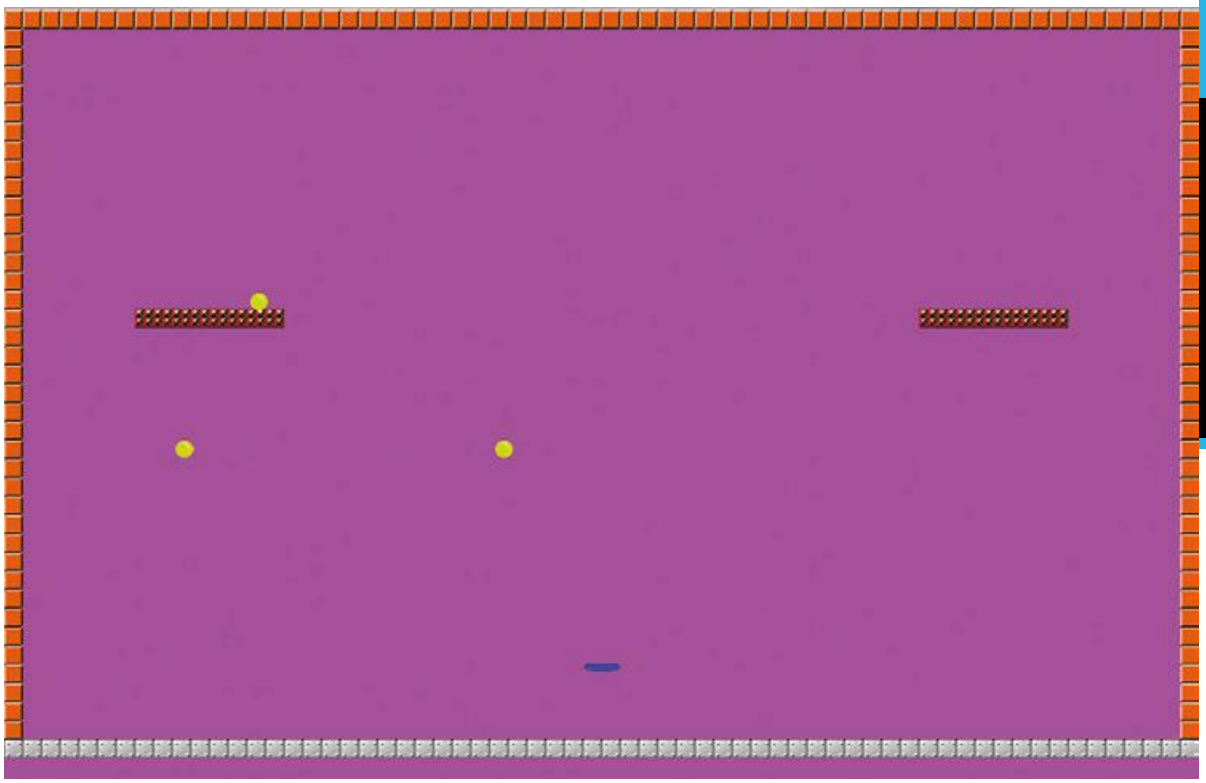
We really don't have to do too much to add sprites to our project. They are basically the same as tiles, but unlike tiles they contain transparent areas so that only the coloured parts are visible, using the alpha to create transparency. All our graphics use red, green, and blue components in their pixel make-up to create colour, but they also have an alpha value that determines how opaque a pixel is.

Sprites are just tiles that move around in pixel steps and have transparency assigned to pixels we don't want to show.

So now we have tiles and sprites, we can create a nice playfield and populate it with our game objects.

Define our game

Before we go forward, we need to define what we're going to do, or we will end up with a mass of unconnected functions. Let's keep it simple and



Language
>C++

NAME:
MagPiHelloTriangle.cpp
SimpleObj.h/cpp
Game.h/cpp
OGL.h/cpp
Ball.h/cpp
NewBall.h/cpp
Paddle.h/cpp

DOWNLOAD:
magpi.cc/zUuMBb

Left
More balls means more obstacles

incorporate some ideas of a bat-and-ball game and also, for fun, shoot things and gather pick-ups.

We will have an unshootable ball which we can bounce around the screen, but if it hits a certain colour of tile it will generate two more shootable balls, which, if shot, will drop pick-ups we can use to increase our score or change our shooting pattern or… hmm, let's keep it simple.

We've now got a nice bat-and-ball game idea, with elements of shooting, that will give us a lot of new code concepts to explore.

Getting some keys

There are a couple of ways to get key-press info in Raspbian, but let's do the easiest, which is just to ask what character has been pressed. There's a simple system call that does that for us, but we do need to set it up. Look at the **`Paddle`** class, which is the only class that needs to test the keys.

Now all we need to do is read the character, check if it's left or right, and move a paddle/bat left or right, preventing it going off the edges.

If we're going to have a bat, we need to have a ball. This an autonomous ball, like our bouncing squares and triangles, but now we will give it some extra conditions for testing. This time we won't test for going to the edge of the screen; we will test if we hit a tile, or if we hit the bat, and change our direction accordingly.

Collision tests

Most 2D collision tests with moving objects are based on two simple ideas: a box check or a circle-to-circle check. These use some simple maths to test if there's an overlap of the shapes on the screen which would indicate collision. Neither system is totally accurate since sprites tend to have a lot of empty space at their edges, making it possible for a box or circle test to detect collision even if no actual visible pixels are overlapping.

But for the most part that's OK. For simple games where objects are moving fairly fast, we probably won't see these false collisions. And if we do, it's fine: we can reduce the size of the box or circle so that it encompasses the bulk of the visible pixels in the sprite. Our ball is naturally round, but we'll encase it in a square which represents its size as a sprite, ignore what will be empty corners, and use a simple box-to-box collision check to decide if we hit.

> "We've now got a nice bat-and-ball game idea, with elements of shooting"

Collision with static objects such as tiles is much easier to do, since we have a grid of our tiles that contains fixed locations; we can test that grid position to see if something is solid and react accordingly.

Creating new things

So far we've been careful to avoid creating new things: almost every object/tile we have used has been stored as a starting instance, and was in place when our program started up. But what if during a game we need

CHECK THE ERRORS

Errors can be confusing. Best to focus on and fix the first couple: fixing those often fixes the others.

POINTERS ARE NOT SCARY

Just remember that it's a variable that points to something useful.

to produce a bullet? We could pre-define it and then switch it on and off; that would work for one or two, but suppose we never actually know how many bullets we need? Or how many times our main ball produces new balls? There's a potentially limitless number, so we need a way to create objects as our game runs, and also to remove them when we are done.

This requires us to think a little about how our Raspberry Pi's memory is arranged. Without getting too technical, there are basically two types of memory that our C/C++ project has available:

> How many times our main ball produces new balls? There's a potentially limitless number

- Stack memory, where all the temporary instances of objects and variables are held.
- Free memory/free store, frequently referred to as heap memory even though that term is deprecated, which is the memory where no data, code or stack currently exists. In other words, it's the memory that is unused when the program is actually loaded and running.

That free memory is available to us, by request! We can use a keyword, 'new', to ask the built-in memory manager for a small object-sized chunk of the free memory where that instance of the object will exist, and we can access it with the much dreaded concept of a pointer. Which isn't as scary as people make it out to be: it's just a variable that points to an object or data which we can then make use of.

Whenever we get data from the memory manager, it blocks off that area of memory. So we have to be careful that when that object has done its work – for example, when a ball is hit and vanishes and is no longer needed – we need to release or delete that memory back to the heap so it can be used again.

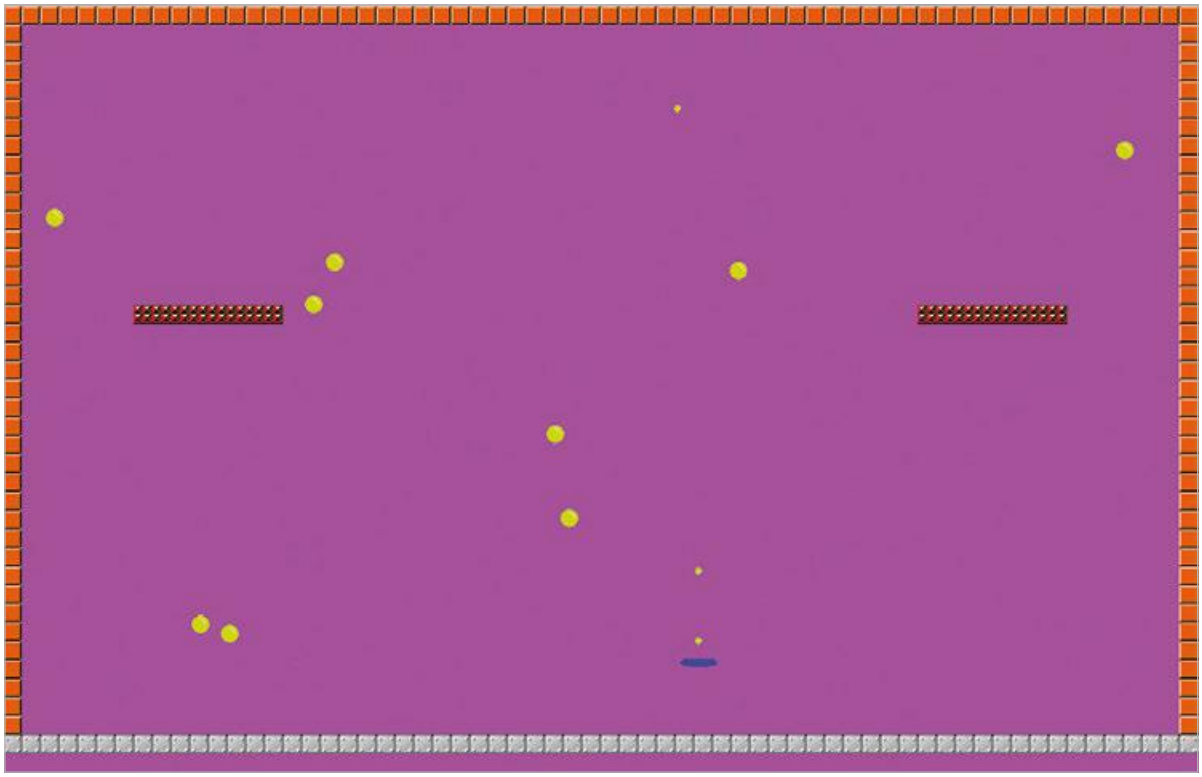
If we don't free memory when it's finished with, but keep asking for new memory, we end up with the dreaded memory leak that programmers worldwide live in fear of. But so long as we make sure every 'new' has a 'delete', we are safe.

Build a game

So let's use what we know already to build a playfield, put a bat and a ball on screen, and when needed, create new balls and bullets – and also when needed, remove them.

All of these objects will be represented on screen by tiles or sprites, but are in themselves different objects doing different things. They do however share at least one feature: a graphic, which in itself contains info on position, and that in turn allows them to be drawn. Also, every object needs to have an update; it may be a complex update, like our player-controlled bat, or it may be simple like the balls bouncing, but an update is needed.

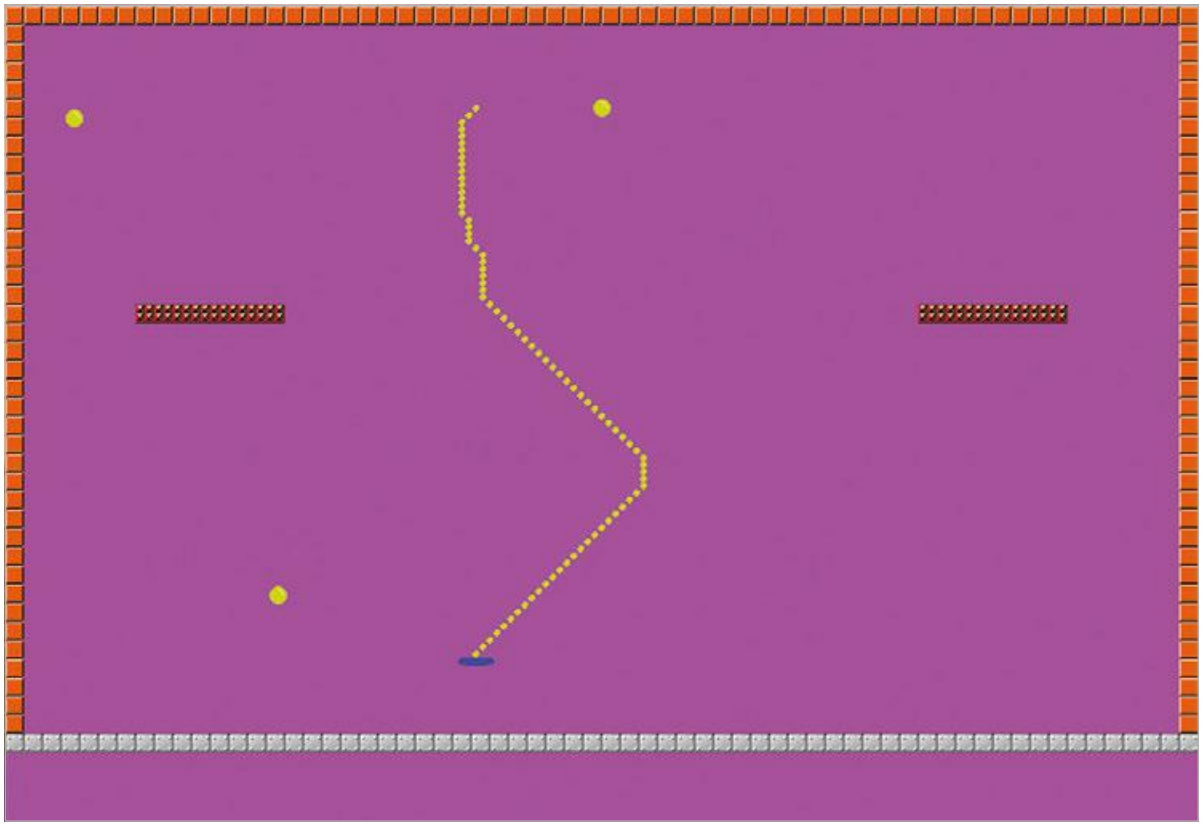
So if everything contains a graphic and has to be drawn and updated, we can gather all the shared concepts to create a base class from which the other types can derive and focus on their differences.



Right There's a lot of balls to shoot, so you'd better get shooting

MEMORY LEAKS ARE BAD

Every 'new' should have a 'delete', to balance the free memory.



KEEP MEMORY TIDY

Make sure your memory is as clean as you found it when you started.

Left
Changing the debounce grants you a hail of bullets

We'll already have a nice base class, called **SimpleObj**, it does not have to change much, but we will make its Update and Draw calls virtual, which means they can be optionally replaced by the classes that inherit it.

The playfield can also be a class, but it's really quite a simple thing: there's no logic needed, the tiles act only as things to bounce off, so all we really need is for them to exist. All that remains is to make a **Paddle** class, which 'derives' from our **SimpleObj**, which means it then has all the **SimpleObj** features but we can add our own update system which will override the **SimpleObj** totally unused update method.

You can see all the definitions for **Paddle**, **Ball**, **Bullet**, and **NewBall** within the source code. They are all very simple things, deriving from **SimpleObj**, and having different update functions, but each operates independently from a core Update and Draw system.

Storing them once we make them

Even though we like arrays very much, they have one important limitation: you need to know the size of them. Since we really don't know how many bullets or balls we will make during the course of our gameplay, we have a problem and need something more flexible.

Time to introduce a storage concept known as a vector. It's basically a type of dynamic array that is able to grow and shrink as the game needs. It is set up a little different to an array, since we must push things into it. Once set up, however, it can be accessed in exactly the same way as an array, which will suit us fine for now.

Fire it up

In the source code, you can see that our **Game** class initialises our play, creates a playfield in a vector, and then new instances of **MyBat** and **MyBall**.

Once initialised we then go into an update and draw cycle which keeps going as long as we want it to (we have not really set a death condition).

We pass the location of the **Game** class to the balls so they can work out if they hit a grid, and we also gain location of the **MyObjects** vector in **Game** class so that we can add items to it. Although passing pointers like this is a little 'clunky', we do get to see C++ doing its party piece very well, where each instance of a class is focusing on itself and its abilities and creating an overall experience with relative ease.

Check out the source code and have a play with the game. Next issue we'll make the collision far more accurate and also start to optimise the untidy shaders we're currently using with some more scary maths.

LEARN TO CODE WITH C

Enjoying using C and C++ and can't wait until next month for more? Check out our Essentials book, *Learn to Code with C*, for more C tutorials for beginners: **magpi.cc/learn-c-book**

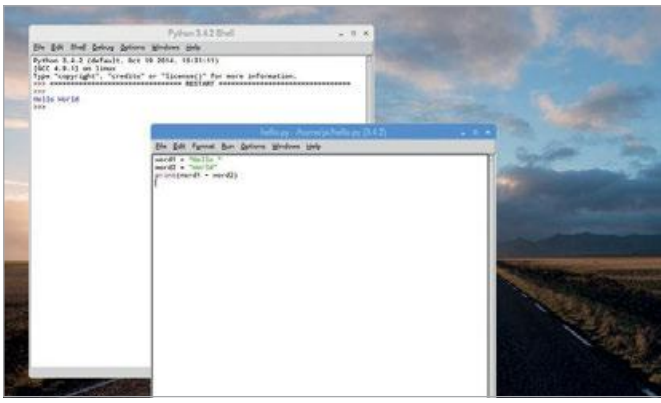
FREQUENTLY ASKED QUESTIONS

NEED A PROBLEM SOLVED?

Email **magpi@raspberrypi.org** or find us on **raspberrypi.org/forums** to feature in a future issue.

Your technical hardware and software problems solved...

CODING ON THE RASPBERRY PI



WHAT CAN I PROGRAM ON THE RASPBERRY PI?

Languages

As it is a fully fledged computer, the Raspberry Pi can run just about any programming language – all you need to do is make sure an environment to run your code is installed on the Raspberry Pi. A lot of common languages already have these environments pre-installed in Raspbian.

Cross-platform

Code you write on the Raspberry Pi will work on other computers – you just need to make sure they have the necessary extra software required, such as Python libraries. Coding on Raspberry Pi helps you learn how to program anywhere.

GPIO

Code that controls the GPIO is not cross-platform, though, as it interacts specifically with the GPIO pins on the Raspberry Pi. It's the same with any exterior input or output from the code: you'll have to recreate that differently on another computer.

WHAT CAN I MAKE ON THE RASPBERRY PI?

Simple scripts

Just want to experiment with coding? The Raspberry Pi is perfect for playing around with bits of Python code, and there are loads of simple resources you can follow to improve your knowledge. You might just be wanting to learn about lists or get hands-on with buttons and LEDs.

Software

Want to make your own software? You can on Raspberry Pi. While you can merely run the software or interfaces you've created as a script, you can also compile them so that they run like normal software.

Games

There are many ways to make games on Raspberry Pi – from simple Scratch games to full-on hand-programmed epics. You need to build up your knowledge and perhaps employ the skills of an artist as well, but the tools are there for you in Raspbian.

WHAT'S A GOOD WAY TO START?

Scratch

Programming languages will differ in how they work. However, they all tend to use a similar structure when being created: 'if' statements that compare things, 'while' loops that repeat actions, and so on. Scratch is a great way to learn the basics of code structure, while also showing instant results from your efforts.

Python

If you want to jump from Scratch to text-based coding, Python is a good place to start. It's designed to be readable – i.e., you can more easily see how it's working – yet still uses the same basic structure as most programming languages. While some consider it an intermediary step, you can still do a lot of programming in Python.

C, Java, and beyond

Once you've mastered Python, you'll have a great, basic understanding of how coding works. From there you can move on to other languages and learn their quirks and structure. Different programming languages are good for different things, and you'll learn along the way how that works.

FROM THE RASPBERRY PI FAQ
RASPBERRYPI.ORG/HELP

WHAT TYPE OF SENSOR DOES THE CAMERA MODULE USE?
The Camera Module V2 has a Sony IMX219 sensor, while the original Camera Module V1 has an Omnivision OV5647. They are comparable to cameras used in mobile phones.

WHICH PICTURE FORMATS ARE SUPPORTED?
The Raspberry Pi Camera Modules support raw capturing (Bayer data direct from the sensor); encoding as JPEG, PNG, GIF, and BMP; and uncompressed YUV and RGB photos. They can record video as H.264, baseline, main, and high-profile formats.

HOW DO I USE THE CAMERA?
There a number of command-line applications provided for stills and video output. These applications provide the typical features you might find on a compact camera, such as set image size, compression quality, exposure mode, and ISO. See the documentation for more details: **magpi.cc/2egdAQA**.

CAN I EXTEND THE RIBBON CABLE?
Yes. We have reports of people using cables up to four metres in length and still receiving acceptable images, though your experience may differ.

WHAT RESOLUTIONS ARE SUPPORTED?
The Camera Module V2 can take photos with a resolution of up to 8 megapixels (8MP). As well as capturing stills, it supports 1080p30, 720p60, and VGA90 video modes. The original Camera Module is capable of taking 5MP photos and can record video at resolutions up to 1080p30.

CAN I HAVE A CAMERA WITH MORE MEGAPIXELS?
No. The Raspberry Pi Camera Modules are the only cameras that are compatible with the Raspberry Pi via its dedicated camera port. There are currently no plans to release a higher-resolution sensor.

HOW MUCH POWER DOES THE CAMERA MODULE USE?
The Raspberry Pi Camera Modules require 250mA to operate. Ensure that your power supply can provide enough power for the connected Camera Module, as well as for the Raspberry Pi itself and any peripherals directly attached to it.

INTRODUCING THE HOTTEST NEW PRODUCTS FOR YOUR PI

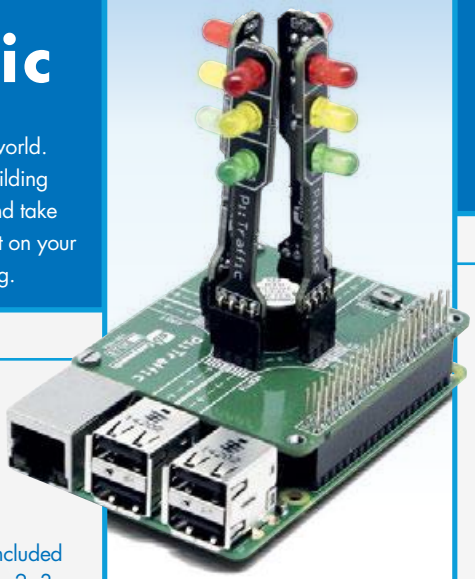
SB Components

THE NEW
PiTraffic

Take your first steps into interfacing with the real world. **PiTraffic** provides the building blocks to explore ideas and take learning further. Just pop it on your Pi and start to learn coding.

FEATURES

- Compatible with Raspberry Pi 3, 2, B+, A+, Zero, and Zero W
- Fully assembled
- Easy Programming
- 40-pin female header included to boost height for Pi B+, 2, 3

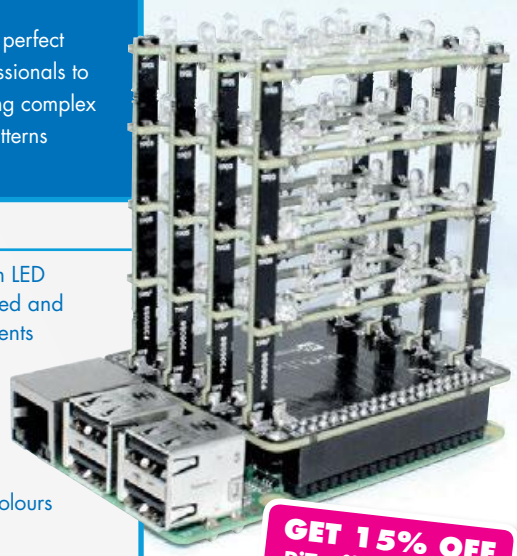


PiCube

PiCube is a 4x4x4 LED Cube perfect for both beginners and professionals to strengthen their logic by typing complex code to draw out different patterns among its various uses.

FEATURES

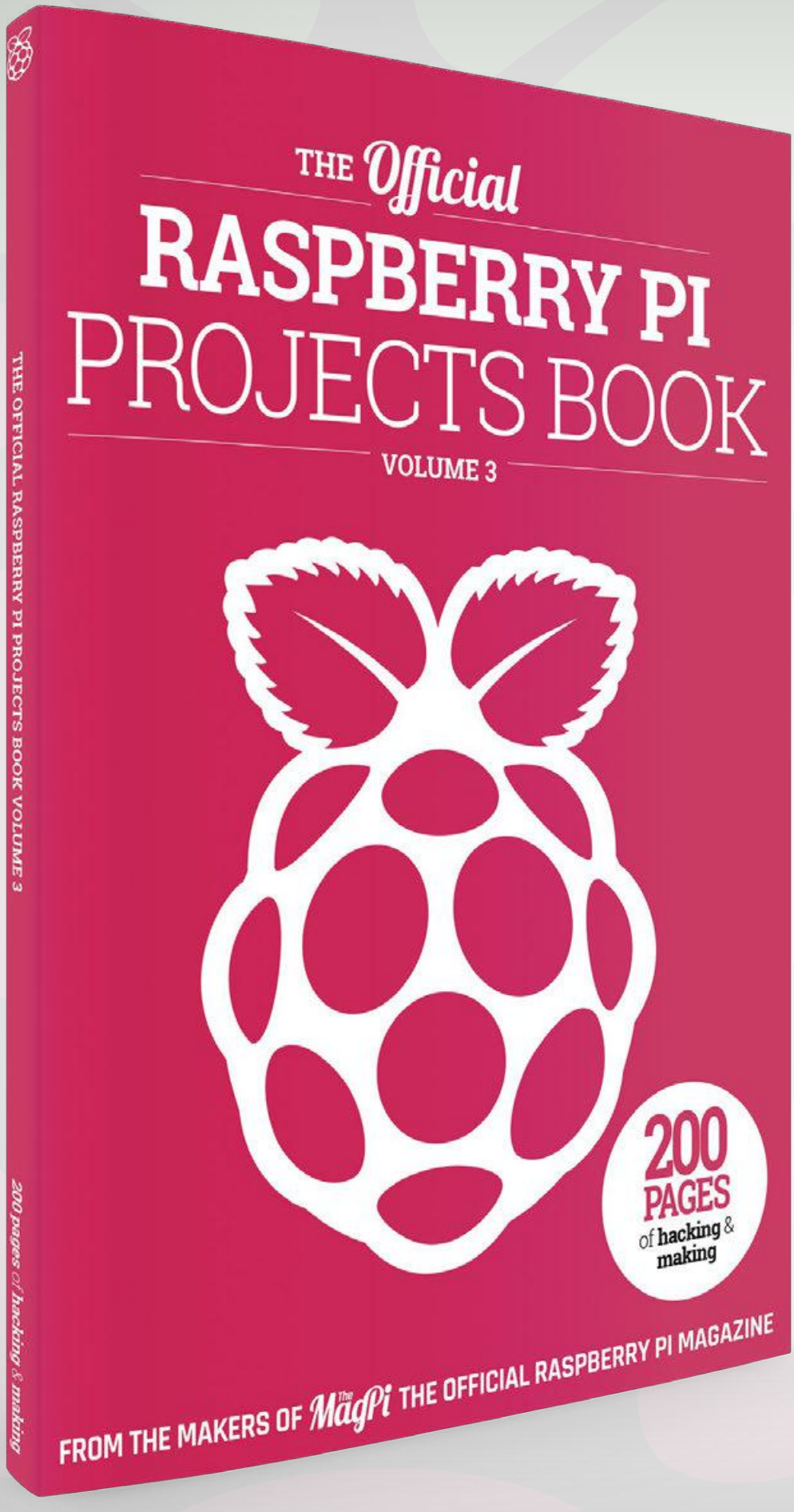
- Each Layer as well as each LED can be individually accessed and controlled as per requirements
- 64 high intensity monochromatic LED's
- 40-pin stacking header for accessing GPIO of RPi
- Available in three vibrant colours RED, GREEN, BLUE
- Comes un assembled and fully assembled



GET 15% OFF PiTraffic & PiCube with this code: MAGPI2017

www.shop.sb-components.co.uk Call: 0203 514 0914

At **SB Components** we strive to offer our customers the best prices for the best products. Our product team works tirelessly to source top quality affordable components from around the world. Raspberry Pi is a trademark of the Raspberry Pi Foundation. Raspberry Pi not included.


THE Official
RASPBERRY PI
PROJECTS BOOK
VOLUME 3
200
PAGES
of hacking &
making
FROM THE MAKERS OF The MagPi THE OFFICIAL RASPBERRY PI MAGAZINE
THE OFFICIAL RASPBERRY PI PROJECTS BOOK VOLUME 3
200 pages of hacking & making

£12.99
200 pages of
Raspberry Pi

THE Official
RASPBERRY PI
PROJECTS BOOK
VOLUME 3

Amazing hacking and making projects
from the makers of The MagPi magazine

Inside:

- How to get started coding on Raspberry Pi
- The most inspirational community projects
- Essential tutorials, guides, and ideas
- Expert reviews and buying advice

Available now
magpi.cc/store
plus all good newsagents and:

WHSmith BARNES&NOBLE

Available on the App Store
GET IT ON Google Play

BUILD A PI ZERO TV STICK

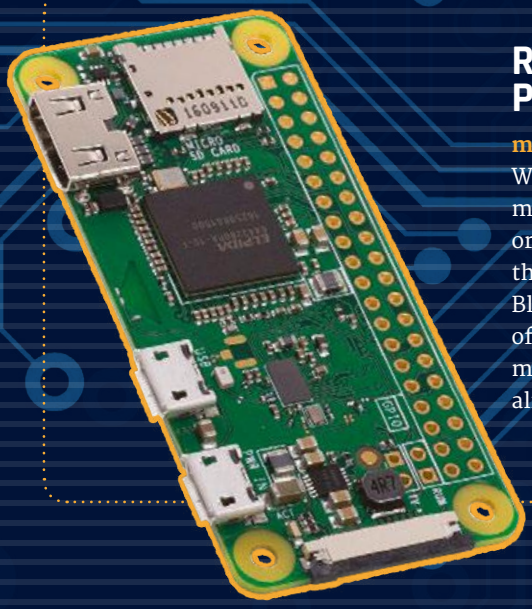
CREATE THE ULTIMATE PORTABLE COMPUTER THAT PLUGS STRAIGHT INTO AN HDMI TV

The versatility of the Raspberry Pi Zero is legendary. Within weeks of its launch we saw it used inside tiny robots, game controllers, and matchboxes to power awesome and fun projects. While the Pi Zero is well suited for these kind of fun experiments, it is still a fully functional, and practical, Raspberry Pi computer.

Let's meld the two together then, and create the ultimate plug 'n' play Pi Zero PC – a Zero Stick. We tasked ourselves with creating a project that could plug straight into your TV like a Chromecast, and only require some external power.

As it turns out, the Pi Zero W is extremely well suited for this. Let's get making.

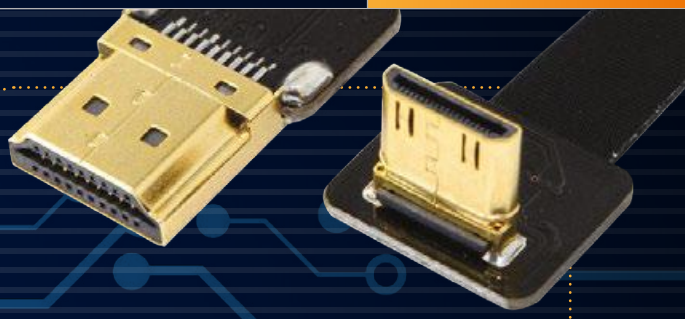
PARTS LIST



RASPBERRY PI ZERO W

magpi.cc/oVqrsl

While you can definitely make this with an original Pi Zero or v1.3, the wireless LAN and Bluetooth capabilities of the Pi Zero W make it much easier to plug in almost anywhere.



MINI HDMI TO HDMI FPV CABLE

magpi.cc/ZiZfAp

Generally used for drones, the slim cable with a bent connector means less space is needed inside the Zero Stick, and also means you don't have to mess around with soldering on a completely different HDMI port. Try to get one with as short a cable as possible.

MICRO USB BREAKOUT BOARD

magpi.cc/MXPzRt

Due to the case dimensions, we need to extend the power ports slightly so that we can actually turn it on. Do it safely with this USB breakout board.

3D-PRINTED CASE

The case for this project was originally created by NODE (**n-o-d-e.net**) as part of their Pi Zero HDMI dongle project: **magpi.cc/rWeSDo**. You can find the STL files here: **magpi.cc/luutuU**.

ELECTRICAL WIRE

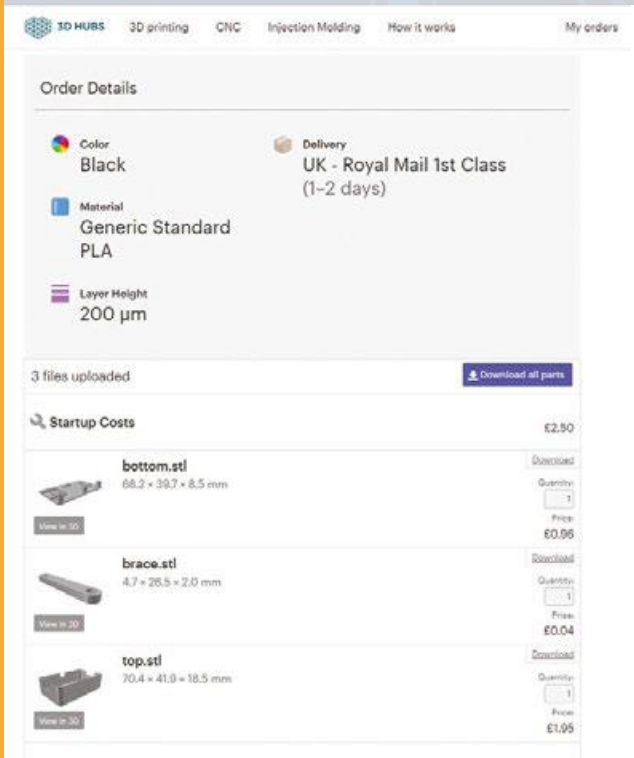
You'll need multicore wiring for soldering on the breakout board, as it's flexible and solders on nice and easy.

SCREWS

To put the case together, you'll need some 5 mm and 12 mm M2.5 screws.

3D PRINTING

Need to 3D-print some parts but don't have access to your own 3D printer, even after our big 3D printing article on page 16? We always like to recommend 3D Hubs (3dhubs.com) – it's a network of thousands of 3D printing companies that will print the case for this project cheaply and quickly. We've even used local printers in the past, so we can go and pick them up ourselves.



ZERO STICK SOFTWARE SETUP

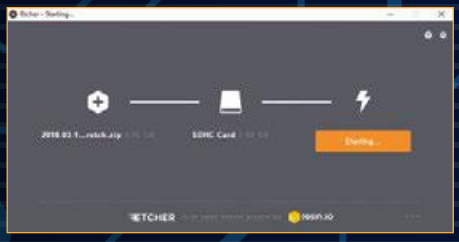
BEFORE WE BUILD THE HARDWARE, LET'S PREPARE THE SOFTWARE

For this guide, we're going to concentrate on Raspbian. This gives us the full PC experience we're looking for, and you can easily work from there to create what you personally desire with your own Zero Stick.

For the initial setup, get your Raspberry Pi Zero hooked up to a monitor and keyboard like you usually would. Once that's done, you won't need to do it again – you can use the Zero Stick as a headless device.

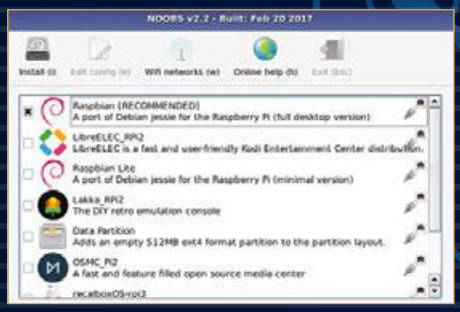
INSTALL RASPBIAN

GET RASPBIAN



You can either download the Raspbian image (**magpi.cc/jSAzxx**), or get the NOOBS image (or a card with it pre-installed) and select Raspbian from its menu. Either way, we like to use Etcher (**magpi.cc/etcher**) to write the image to the microSD card. Plug the microSD card into your Pi Zero and get it hooked up to your (temporary) setup so we can configure it.

INITIAL CONFIGURATION


NOOBS v2.2 - Built: Feb 20 2017
Raspbian [RECOMMENDED]
A port of Debian jessie for the Raspberry Pi (full desktop version)
LibreELEC_RPi2
Raspbian Lite
A port of Debian jessie for the Raspberry Pi (minimal version)
Lakka_RPi2
The DIY retro emulation console
Data Partition
Adds an empty 512MB ext4 format partition to the partition layout.
OSMC_Pi2
A fast and feature filled open source media center

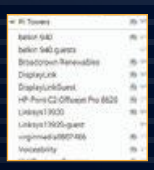
Turn the power on. If you're using NOOBS, select the full version of Raspbian and let it perform the installation. If you've burned Raspbian directly via Etcher, it will resize the partition so that Raspbian is using the full SD card. After all that, it will reboot, and you will load into the Raspberry Pi Desktop.

TRANSFER ANY FILES (OPTIONAL)

While you have your Pi Zero hooked up and more accessible, now is a good time to get any documents or program files loaded onto it. You'll be able to transfer them later over a network connection anyway, however.

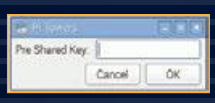
WIRELESS INTERNET

FIND YOUR WIRELESS NETWORK



While you may not be making the Zero Stick for use around the house, it's useful to get it set up on your home WiFi network for testing and tweaking. Click the wireless LAN symbol on the top right of the taskbar. It will bring up a list of wireless networks you can connect to – find yours and click on it.

ENTER THE PASSWORD



If your wireless network requires a password, the password prompt will show up. Enter it, making sure to use the correct upper- and lower-case letters, and press OK. It will save the password for the future, and it will even work if you boot into the command line instead of the desktop. You'll be able to add more wireless details in the future, so you'll be fine to use the Zero Stick elsewhere.

INSTALL EXTRA SOFTWARE

If you have any software or Python libraries you know you'll be needing for your Zero Stick, now's a great time to get them pre-installed and ready for testing when you complete the stick. You'll be able to install extra software once completely set up anyway.

CONNECT A KEYBOARD AND MOUSE

Thanks to the Pi Zero W's radio chip, you can easily connect a Bluetooth keyboard and mouse to the Zero Stick. Fewer cables means better portability.

KEYBOARD

Click on the Bluetooth icon at the top right of the screen. Click on 'Set up New Device' and then click Next to start scanning. Turn your keyboard on and make it discoverable (you may need to check the manual). It should then show up in the list of items while the Raspberry Pi scans.

Click on it and you may be asked to create a passkey. Enter whatever you wish, but make sure to remember it as you'll then be asked to enter it on the keyboard. Once you've done that, pairing will be finished.

MOUSE

Similarly to a keyboard, you'll need to click on the Bluetooth icon and go to 'Set up New Device' to get started. Make the mouse discoverable and connect to it. Unlike the keyboard, the mouse will likely have a preset passcode to connect – check the manual for your mouse to find out what it is and enter it into the prompt on your Raspberry Pi.

Once paired, it will automatically pair next time you turn the Raspberry Pi on.

REMOTE CONTROL WITH SSH

Fancy a different method of controlling your Zero Stick? Here are some alternative remote control and access methods that might be useful to you...

As long as you're on the same network as the Zero Stick, you'll be able to connect to it from another PC. That defeats the purpose of the project's portability, though, so instead we suggest using an SSH app on your smartphone to connect.

Either way, you'll need to activate SSH in Raspbian before connecting to it. Open Raspberry Pi Configuration by clicking on the Raspberry Pi logo at the top left of the screen and then going to the Preferences menu.

Go to the Interface tab and click Enable for SSH, then reboot. To connect to your Zero Stick, all you'll have to do is point your SSH app towards **pi@raspberrypi** and enter the password '**raspberry**'.

BUILD YOUR ZERO STICK

BUILD THE ULTIMATE PORTABLE PC

TOOLS

- Soldering iron (plus accessories)
- Small screwdriver
- Hot-glue gun

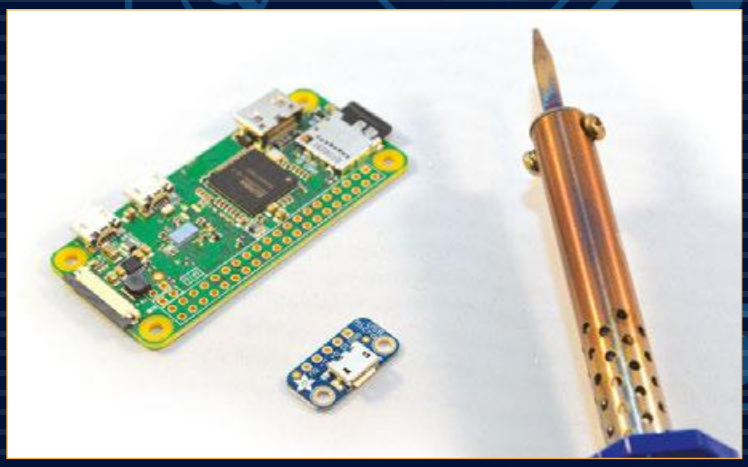
GET THE 3D PARTS

The 3D-printed case is essential for keeping this build neat, tidy, and portable. As it's small and will be used generally out of sight, you can make it any colour you wish and use any plastic type to build it. We did ours in black ABS and it came out just right.

SOLDER ON THE BREAKOUT BOARD

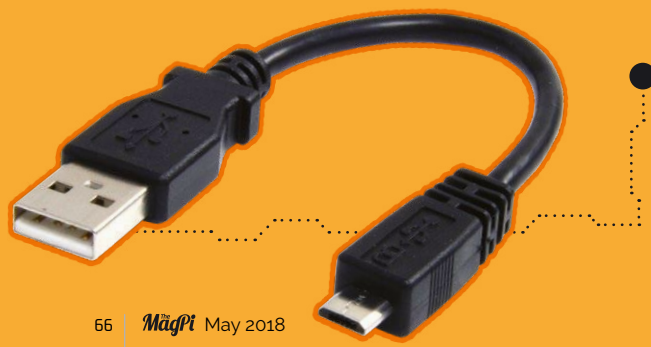
The micro USB breakout can be soldered to either the pads under the USB power socket, or directly to the correct GPIO pins. The breakout board makes sure we're using a USB power supply so that the voltage will be correct (5 V). Solder a wire between the 5 V hole on the breakout board and physical pin 4 on the GPIO, and solder a ground wire between physical pin 6 and the GND hole on the board.

If you want to be extra safe, put a hot-glue gun onto a low setting and dab a little glue over the solder.



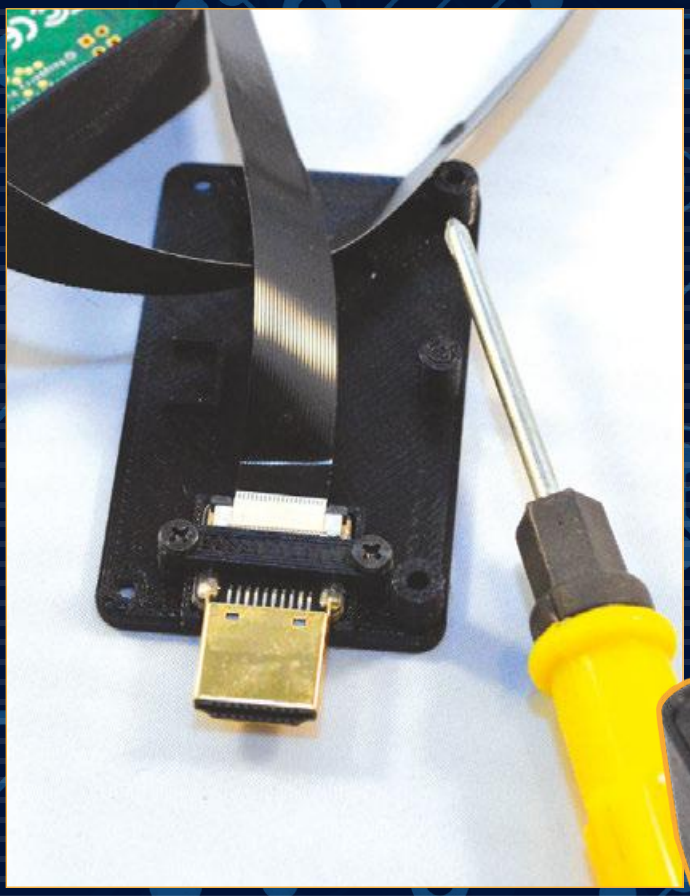
POWER YOUR ZERO STICK

What is the best option to power your project?



TV USB PORT

This seems like a no-brainer option, right? You're already plugging your Zero Stick into a TV's HDMI port, so why not keep the cabling to a minimum by powering it from your TV? While this may work, it's more than likely that the TV would not provide enough power to the Zero Stick – in that case you'll see a lightning bolt appear in the top right of the screen.



> Test it all out on the back of your TV or any other display that supports HDMI

SECURING THE CASE

You'll need to fold up the remaining cabling of the FPV cable and then put both parts of the case together. Using the longer screws, secure the two parts together. You're done!

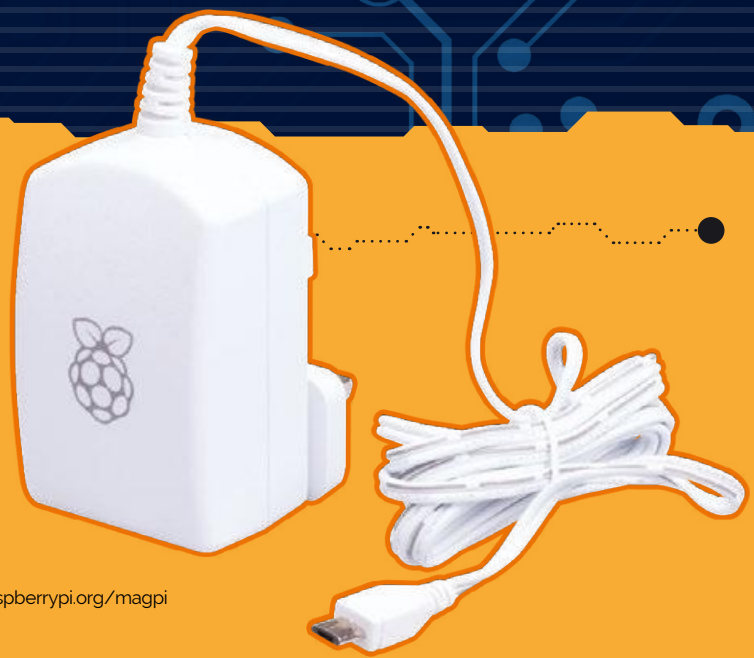
Test it all out on the back of your TV (or any other display that supports HDMI input).

ASSEMBLE THE ZERO STICK

Glue the micro USB breakout board to the opening on the side of the case. Rest the Pi Zero in its slots inside the large piece of the case and make sure your wires are long enough.

Plug the FPV connector into the Pi Zero and then put the Zero back into position and screw it down to hold it in place.

Position the other end of the HDMI cable and secure it with the little bracket part we printed.



POWER SUPPLY

An official power supply or a sufficiently powerful mobile charger will be the best thing to use. This way the Zero Stick will be guaranteed to get enough power to run at full speed – just make sure there's a mains socket handy!

If all else fails, try a mobile battery! Make sure it outputs at least 2 A, though

DO MORE WITH YOUR ZERO STICK

USING IT AS A PC IS ONLY THE BEGINNING

The Raspberry Pi itself is very flexible in what it can do, so why not our Zero Stick? Here are some suggestions for other great ways to use it, some of which would be excellent projects on their own.

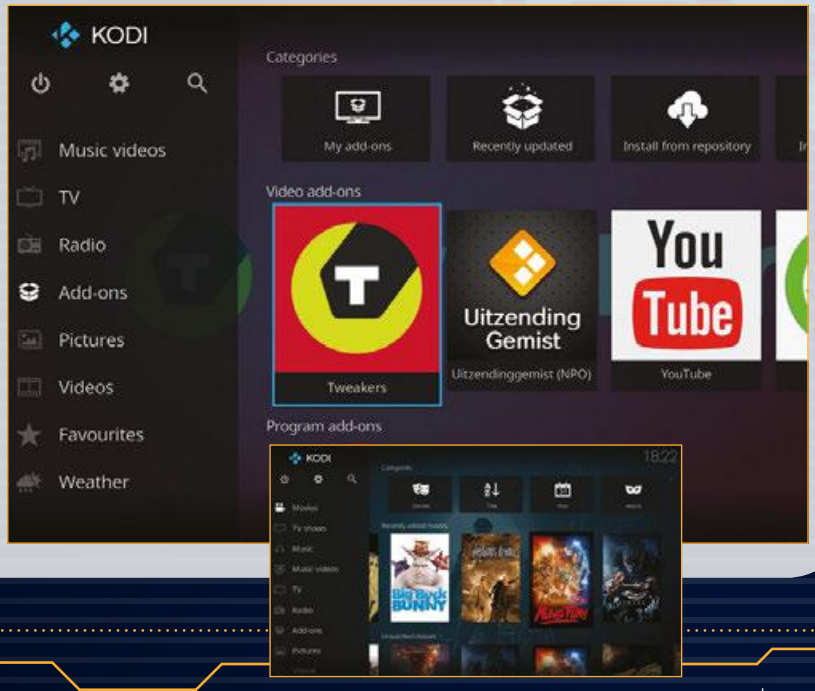
TV STICK

At *The MagPi*, we always make sure we have a spare Raspberry Pi running Kodi. While the draw of Chromecast is strong, the Zero Stick with Kodi installed becomes the perfect Pi-powered Chromecast alternative.

We still recommend LibreELEC (**libreelec.tv**) for making a Kodi box with Raspberry Pi. Grab the Pi Zero-compatible image and burn it to a microSD card. Perform the basic setup and configuration before putting the Zero in the case, as before. You can use a Bluetooth remote control or simply a Kodi app on your smartphone to control the system, so you don't need to sort out any IR dongles.

TIPS

- Want some media on there at all times? Try using a bigger microSD card.
- If you want to use it outside your house, try connecting it to a smartphone's hotspot beforehand for easy remote access.
- You don't have to use Kodi for this: any Pi Zero-compatible HTPC software will work.



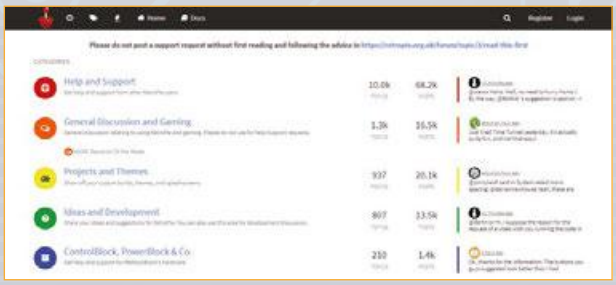
GAMES STICK

If you've been able to fit a Pi Zero into something, someone has turned it into a games console. The Zero Stick is no different, only requiring some different software to get it playing some games.

RetroPie (**retropie.org.uk**) is a great start, and it does have some support for Bluetooth-connected controllers as well. You will definitely need to set it up with a more traditional wired input first, though, before you can get the Bluetooth controller support working (**magpi.cc/YaXMHN**).

TIPS

- There's an opening in the rear of the case if you fancy adding a USB port for a USB controller.
- You can also turn it into a Minecraft stick; just have Minecraft Pi start at boot.



WEB STICK

Web kiosks are an important use case for small PCs, especially tiny ones like the Raspberry Pi. Using Raspbian or a dedicated web kiosk OS like Raspberry WebKiosk (**magpi.cc/hWlxwW**) on your Zero Stick, you can create a low-cost, secure web kiosk PC that easily hides away behind TVs.

While you may not run a hotel or student union that could make use of it, it's not a bad idea for a guest room or a safe way for younger kids to browse the internet at home.

Make sure to change the default user names, passwords, and permissions if you plan to adapt Raspbian.

TIPS

- A monitor will do for this as you don't necessarily need sound.
- If you do require sound, though, Bluetooth speakers will also work.

PEN-TEST STICK

If you work in IT, you sometimes need to do some pen-testing of your network. While you'll probably need something a little more beefy if you want to get serious, you can't go wrong with some quick tests with the Zero Stick that fits in your pocket.

Kali is one of the best-known pen-testing Linux distros, and luckily there is a version of it for Raspberry Pi (**magpi.cc/twNmnd**). Install it as usual and set it up in your preferred manner – you may need to add some extras to it, so feel very free to customise it.

TIPS

- Modify the STL so you can add it to your key fob.
- You can get micro USB to Ethernet adapters.
- Don't use it for evil.

RASPAD

A premium Raspberry Pi tablet that does all the building work for you. **Rob Zwetsloot** investigates the RasPad...

Raspberry Pi tablets are one of those Pi projects that took a little while to catch on in the early days, as the hobbyist electronics suppliers had to catch up a bit with the potential of the Raspberry Pi after its runaway success. It's been a few years now since people have been able to create their own tablet-esque Pi computers, made easier with the introduction of 7-inch touchscreen displays and such.

The RasPad aims to cut out the DIY part and leave you with a functional, very usable, Raspberry Pi tablet. The only construction you need to do is slot in the Raspberry Pi – or one of many other popular microcomputers/single-board computers. The finished product looks and feels great; it's solid and has a decent heft to it. The big case makes it pretty easy to hold while also doubling as a way to angle the screen on your table towards you.

Like any good tablet computer, the RasPad has an internal battery which makes it portable. As there's no battery indicator in Raspbian, battery life is handled via some LEDs on the bottom of the case – a more visual reminder that in our opinion works a little better than a normal tablet or laptop percentage.

It's possibly a little too heavy to use in some traditional tablet capacities, though, especially with it using desktop software over smartphone apps; you won't be using it to catch up on Twitter in bed in the morning, for example.

A tablet for makers

While that's a bit of a shame, it's not really designed for an early-morning social media catch-up. What it is designed to do is give you a bit of a head-start with using a Raspberry Pi to make some creative projects.

The microSD card comes pre-installed with a compatible version of Raspbian, and even in its case the Raspberry Pi Camera Module connector and GPIO pins are easily accessible (although using a ribbon cable instead of individual jumper cables works a bit better).

Performance on such a device is incredibly important – after all,

Related

TOUCH-SCREEN DISPLAY CASE

It's cheaper to get a Raspberry Pi 7-inch Touchscreen Display (£60) and put it in this case, although you don't get anywhere near as many options.

£15 / $21

magpi.cc/NUaySe

magpi.cc/kPbzQN

£205 / $290

Maker Says

"A Raspberry Pi tablet to build your creative projects"

SunFounder

you're going to get a bit frustrated using the RasPad for projects if it's laggy and slow. We've been pleasantly surprised just by how silky smooth it runs; while the Raspberry Pi Desktop isn't exactly optimised for touchscreens, on a screen this size (10.1-inch) it works very well, with no discernible lag we could discover even with a few resource-heavy tabs open.

The on-screen keyboard is not amazing, though, as the team have to work with what they have – but it works well enough. Even then, you can always plug in a USB keyboard (and mouse!) so you can use Raspbian in a more traditional manner.

> "We've been pleasantly surprised just by how silky smooth it runs"

Display choices

The RasPad also takes into consideration that you might want to output the signal to a full-sized monitor, letting you connect a HDMI cable to it and bypass the main screen entirely. While this is excellent for giving you options, we feel it could have done with one more orientation mode so that you can rest it on the back of the device so it can be more upright. Currently the angle is not really deep enough for that – plus you'd need to rotate the screen manually – but it seems a bit like a no-brainer with the way it's designed. Still, outputting to a TV does partly make up for that.

The battery life is pretty good on the system as well, even with the screen on. As the Raspberry Pi eats up very little power while idle, the tablet won't run down within an hour with the screen off, even when you've inserted a relatively more power-hungry Pi like a 3B+. We've also been told the power system will be improving for the full release as well, which is great to hear.

Making with the RasPad

Actually using the RasPad is a great experience. For Scratch, the touchscreen works well, but we'd recommend a USB keyboard for Python or other kinds of programming. While you'll rarely see the RasPad installed in a project (unless it needs the screen), you can at least do the setup with it plugged in before removing it from its case. We can even envisage doing some maintenance on the go by returning the Raspberry Pi to the RasPad case, thanks to the quick way you can detach and attach the board.

It's a really fun device that makes excellent use of the Raspberry Pi, and we can definitely see ourselves using it more in the future.

Last word

A great way of getting started with Raspberry Pi, this tablet is well built and comes with plenty of options that make it very useful to makers of any skill level.

★★★★★

magpi.cc/ptbHkP

£50 / $55

Maker Says

"Learn how to create code that can control three exciting projects"

Maker Life

GETTING YOU STARTED KIT WITH RASPBERRY PI ZERO W

Everything needed to get started with physical computing. By **Phil King**

Maker Life's range of electronics kits aim to provide a plug-and-play learning experience with no need for soldering. Such is the case with this starter kit, which includes a Raspberry Pi W with pre-soldered GPIO pins. Also supplied in the box are a 16 GB SanDisk microSD card with NOOBS pre-installed, USB and HDMI adapters, and a USB hub to connect your own keyboard and mouse. At just 1 m long, the included HDMI cable may prove too short for some. There's no power supply either – instead, the user is advised to connect the supplied USB adapter cable to a PC.

As this is a physical computing kit, the box contains a selection of electronic components: five LEDs, five 220 Ω resistors, two push-buttons, ten male-to-female jumper leads, and a mini 170-point breadboard. We'd have preferred at least a half-size, 400-point breadboard since the mini one lacks ground/power rails, so requires a separate ground pin to be used for each component in a circuit.

Instructions included

A glossy 44-page instruction booklet explains all the components, how to connect the Raspberry Pi, and how to install Raspbian from the NOOBS menu. Following a very brief one-page guide to the Raspbian desktop, it then moves straight on to coding three physical computing projects in Python: 'traffic lights' (lighting three LEDs in sequence), controlling an LED with a push-button, then putting those two concepts together to make a pedestrian crossing – a shame there's no buzzer for a beeping sound, though.

While clear wiring diagrams and tables are provided, you keep having to refer to the GPIO diagram on an earlier page to find the correct pins. Also, there's no explanation of how the solderless breadboard is wired, or why the resistors are required.

The Python programs – which use the GPIO Zero library for simplicity – are much better explained, broken down into steps and with a full code listing at the end.

Last word

With around £43 worth of components (by our reckoning) plus a 44-page booklet, this kit represents reasonable value for money in a convenient package. It has a few shortcomings, however, and does throw complete Raspberry Pi newcomers in at the deep end, straight into physical computing.

★★★☆☆

Related

CAMJAM EDUKIT 1

A great value-for-money starter kit for physical computing, including LEDs, a push-button, piezo buzzer, jumper wires, resistors, and a 400-point breadboard.

£5 / $6

magpi.cc/drAnUT

CoderDojo

Can I start a CoderDojo club in my local area?

CoderDojo is a global network of free, volunteer-led, project-based programming clubs for children aged 7–17. Dojos are championed by individuals all around the world who are passionate about giving young people the opportunity to learn to code.

Starting a Dojo is a fun and incredibly rewarding experience

You don't need to possess technical skills to start a Dojo. The most important attribute is that you can bring people together for a shared goal.

We're ready to support you by providing:

- **Learning resources and guides**
- **A free event management system**
- **Certificate templates, digital badges, and giveaways**

"I started a Dojo to give my kids a place to meet other children also interested in programming and making games. I get to see them making new friends, learning from one other, and they loved it. Realising how I had created such a wonderful place for children has ignited a spark in me."

- Maroes, CoderDojo NL

Start your own club. Join us at CoderDojo.com

The CoderDojo Foundation is Irish Registered Charity (524255) CHY20812
and is part of the Raspberry Pi Foundation, UK registered charity 1129409

Part of Raspberry Pi

magpi.cc/yisLUK

£43 / $50

Maker Says

"Any developer can add clean audio to their application"

Andrea Electronics

ANDREA PUREAUDIO ARRAY MICROPHONE DEVELOPMENT KIT

Promising performance, but accessible to professional developers only

Billed as 'the ultimate Raspberry Pi 3 microphone', the Andrea PureAudio Array Microphone Development Kit (MDK) is a three-part product. The first is an Andrea SuperBeam stereo microphone. The second is a PureAudio USB sound card with 3.5mm microphone and speaker jacks, without which the software – the third part of the bundle – won't run.

The USB sound card is used as a hardware token to prevent unauthorised redistribution of the software. It does not, however, provide special processing functionality.

The absence of hardware implementation has an impact on performance: while recording or listening for its activation phrase, the sample application caused a 60 percent increase in power draw and a sustained 30–40 percent single-core CPU load – a big impact for anyone working on an embedded design.

Audio filtering

The primary feature of the MDK is an audio filter library which uses Andrea's various technologies to improve the performance of the microphone in various conditions. Active noise cancellation drops background noise considerably, a beamforming mode helps when the speaker is off-centre from the microphone, and echo cancellation works to prevent speaker audio intruding on the recording. All these work well.

However, software must be specifically built to use the library. The only software that makes use of it at present is the extremely limited PureAudioPi sample application included in the bundle as source code, which listens for the preset key-phrase "Hello Blue Genie", then triggers a ten-second recording.

The functionality of the kit can be extended, at a cost: add-on 'vocabulary packs' are available for £9/$10 each. These add the ability to recognise simple commands such as 'up', 'down', or 'today's weather', but they come with no built-in functionality beyond printing confirmation to the Terminal, and you can't build your own vocabulary packs either.

Last word

While the filter library is smart, being tied to a specific USB sound card and requiring significant resources make the MDK hard to recommend.

★★☆☆☆

Related

AIY VOICE KIT V2

For those looking for a more off-the-shelf solution to voice control, the AIY Voice Kit V2 is a great choice that now includes the Pi Zero WH.

£TBC / $50

magpi.cc/fmVcQV

HackSpace

TECHNOLOGY IN YOUR HANDS

THE NEW MAGAZINE FOR THE MODERN MAKER

SUBSCRIBE AND SAVE UP TO 35% on the cover price

ISSUE #06

OUT NOW

hsmag.cc

Available on the App Store

GET IT ON Google Play

Maker Says

"The world's smallest and lightest personal computer"

Ashish Gulhati

NOODLE PI

A special 3D-printed kit that turns your Raspberry Pi into a tiny and very portable computer. **Rob Zwetsloot** takes it for a spin

A successful Kickstarter campaign, the Noodle Pi is a special case designed to house a Pi Zero, a Pimoroni HyperPixel display (**magpi.cc/2ueAs8L**), a battery, and various other little bits of electronics to bring it all together. For £35 you get the 3D-printed case/parts, some wires, a Camera Module cable adapter for the Pi Zero, and a link to a list of the rest of the electronics that you need. You'll then have to put the work in to build the system and create your little Noodle Pi box.

With the case, the Pi Zero, and the extra parts, you're looking at about £100 to put the whole thing together, although if you have £140 in Bitcoin you can buy a pre-assembled version. The latter comes with a few extra bits as well, to make use of the dock system that allows you to attach it to your belt, or to a watch strap for that retro cyberpunk flair.

To solder or not to solder

Building the system requires no soldering, although if you prefer to solder a GPIO header onto a Pi Zero W instead of using the hammer header system from Pimoroni or buying a Zero WH, you can do so. As most of the Noodle Pi uses hardware already designed to connect to the Raspberry Pi (such as the HyperPixel which acts as a HAT, or the Camera Module whose cable slots into the CSI port), the only thing you'd need to solder is the PowerBoost charger and battery.

The solution provided with the kit involves pre-prepared wires that are trimmed long enough that you can loop them around the edge of the PCB and twist them together. It also makes use of the holes in the GPIO header on the HyperPixel to wedge the wires into the correct position.

While it works, the designer admits that it is a bit tricky. If you have a soldering iron handy, we suggest using that anyway – it's only a couple of wires and a switch, after all.

Sturdy build

Once all complete, the Noodle Pi feels nice and solid, with enough heft to make it feel like a decent bit of kit. Think in terms of a small smartphone. It's this size that enables it to be holstered to your

Related

RASPAD

A much larger portable Pi computer with more of an emphasis on learning through electronic projects.

£205 / $290

magpi.cc/kPbzQN

noodlepi.com

£35 / $49

belt or used with a 3D-printed adapter to connect to your wrist. It's a little clumsy attached as a wrist computer to be honest, but it's a neat party trick even if we have seen Fallout Pip-Boy builds do a similar thing but a little better.

'dock' specific types of Bluetooth keyboards to the case so it's a bit sturdier and closer to the screen.

Mobile operations

Using it as a portable PC is a satisfying experience. The battery lasts for a fair amount of time

> "It's a little clumsy attached as a wrist computer to be honest, but it's a neat party trick"

While it runs smoothly with the Pi Zero W hardware, the touchscreen doesn't include any touchscreen keyboard support by default. The HyperPixel allows you to do a lot, but for a full 'desktop experience' you'll need to attach a keyboard and mouse. You can do this via Bluetooth, and you can also get special add-on parts to

and with the relatively high resolution of the display there's enough screen real-estate to use it for some simple tasks. With the GPIO pins inaccessible due to the necessity of the build, you're a little more limited in the kinds of projects you can do with it – although the mini HDMI out and the USB data port are readily accessible so there's definitely some level of extra input/output you can add to the Noodle Pi.

It's also fairly flexible – you can install RetroPie on a spare SD card and use a controller on it for some retro gaming. You could even use it as a portable Kodi machine to plug into other TVs, although there are better, cheaper, and easier ways to do that.

It's a great little project with some interesting use cases – and unlike a few of the projects that we cover, you can easily make it yourself!

Last word

A simple case kit that lets you put together a pretty cool computer, the Noodle Pi is an interesting product that should appeal greatly to a number of makers and hackers.

★★★★☆

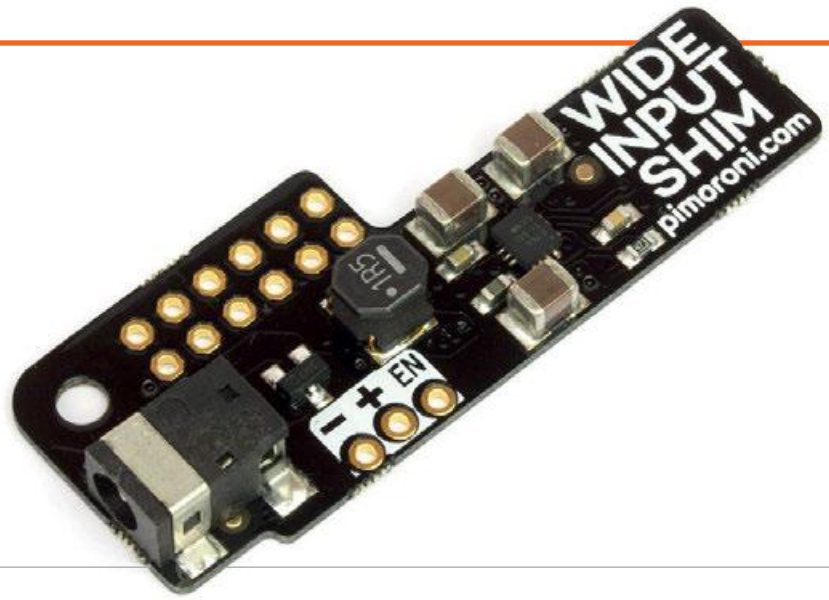
magpi.cc/FLKejY

£10 / $14

Maker Says

"Power your Pi with a wide range of input voltages, from 3 to 16 V!"

Pimoroni



WIDE INPUT SHIM

Lucy Hattersley looks at expanding power supply options with this dinky board

The Wide Input SHIM from Pimoroni is a handy board that widens the range of power sources you can use with a Raspberry Pi.

The bare Raspberry Pi board is very particular about its power source. It likes a 5.1 V micro USB power supply – the kind commonly used by smartphones.

This 5.1 V requirement is enough to power the board, most HATs, and even the occasional servo. But what if you want to integrate the Raspberry Pi within a larger project with different voltage requirements?

That's where the Wide Input SHIM steps in. With it you can use a range of power supplies, from 3–16 V. These are converted by the small board to 5 V with 2 A output thanks to a TPS63070 buck-boost converter chip (**magpi.cc/QZCjmH**).

Measuring just 45×17×6 mm (L×W×H), the SHIM tucks neatly into the corner of the Raspberry Pi board and takes up the first 12 of the 20 GPIO pins. It's also super-skinny, at just 0.8 mm thick.

> "The SHIM tucks neatly into the corner of the Raspberry Pi"

Connection options

You can solder the board to the supplied female header and pop it to the end of the GPIO pins. Or, if you're feeling adventurous, you can solder the board directly to the GPIO pins on your Raspberry Pi. It's not a soldering project for the faint of heart (or wobbly of hand).

The board features a 3.5 mm barrel jack connector, and included in the package is a 5.5 mm adapter cable. There are additional + and – pins on the SHIM so you can wire up power directly. Or you can use the 3.5 mm to connect black and red tinned wires to integrate the SHIM with your project or hook it up to a bench power supply.

Next to the + and – pins is an EN pin. Pulling this to ground will cut the 5 V voltage output. Be warned though that it won't perform a clean shutdown. There is also a distinct lack of an on/off button on the board. We would have liked to have seen the button and safe shutdown script from the OnOff SHIM (**magpi.cc/itJzUx**).

Related

NANOMESHER HACKABLE RASPBERRY PI SWITCH

A programmable power switch for your Raspberry Pi that can be controlled from a button or an IR remote.

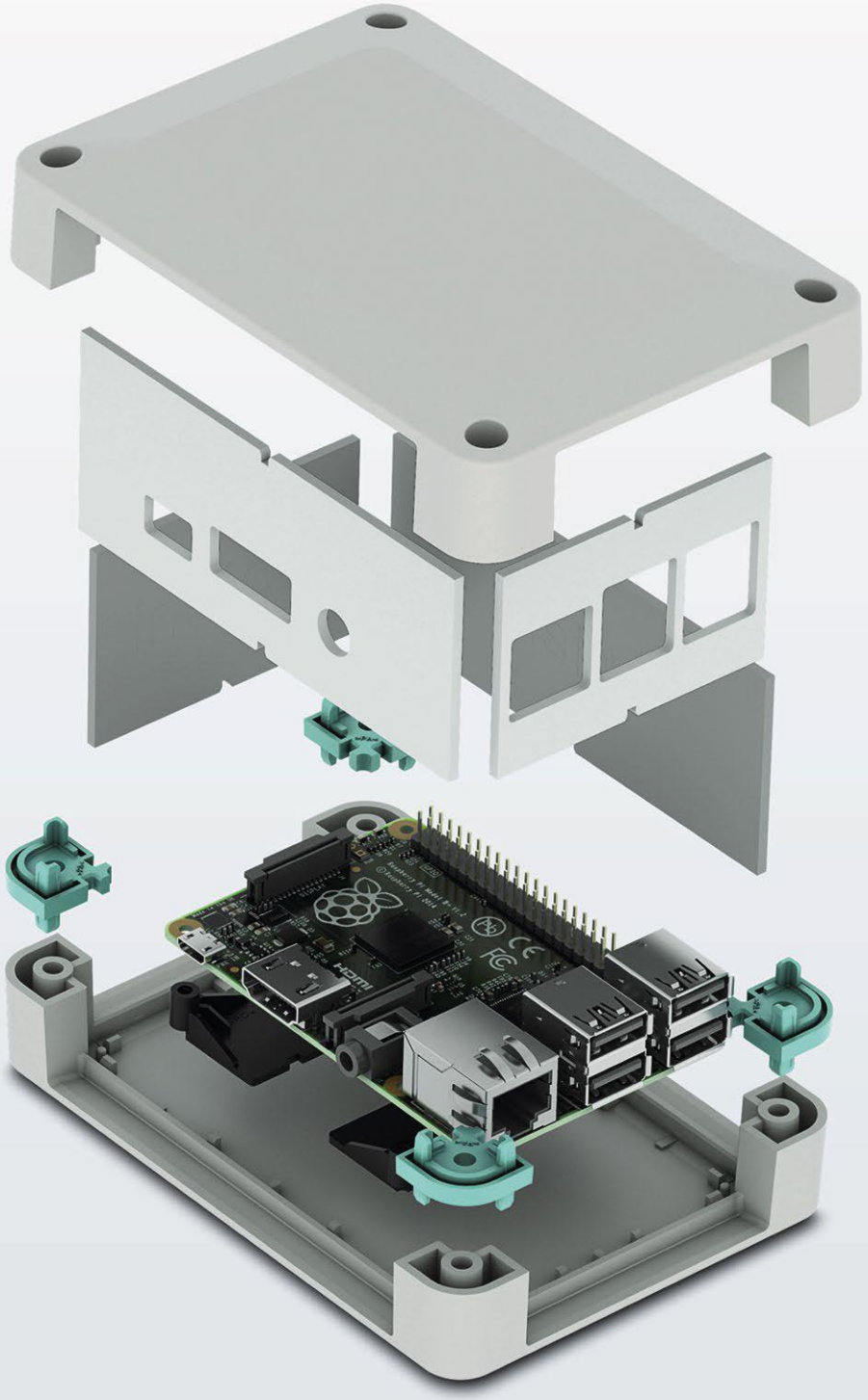
£22 / $31

magpi.cc/wokNaG

Last word

A perfectly good product that lets you use a wider range of power inputs in your project. It's fiddly to solder directly onto the board, but the supplied female header makes for an easier solder project. We would have liked an on/off switch, though.

★★★★☆



Raspberry Pi 3 B+ finds a new home

Versatile enclosure for Raspberry Pi 3 B+

The new UCS-RPI Universal Case System is compatible with the recently launched Raspberry Pi model 3 B+. It has pre-milled side walls for easy access to the I/O and power inputs and is available in black or grey and two sizes. Complete with glue dot location posts to secure the single board computer to the case.

For additional information call 0845 881 2222 or visit

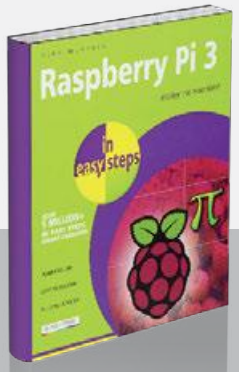
phoenixcontact.co.uk/UCS-RPI

PHOENIX CONTACT
INSPIRING INNOVATIONS

RASPBERRY PI BESTSELLERS IN EASY STEPS

Whatever the tech topic, there is a book that breaks it down into easy steps!

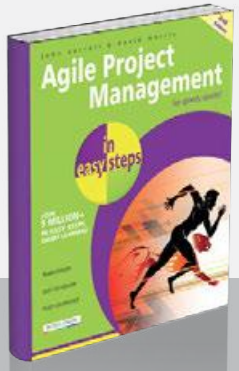
RASPBERRY PI 3 IN EASY STEPS



Author: Mike McGrath
Publisher: In Easy Steps
Price: £10.99
ISBN: 978-1840787290
magpi.cc/XWXdip

Useful Pi intro that manages to touch on media players, shell scripting, and animations in Scratch, before turning into a handy Python programming intro – with graphics, games, and some GPIO projects as well.

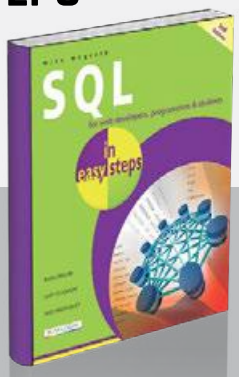
AGILE PROJECT MANAGEMENT IN EASY STEPS



Authors: John Carroll, David Morris
Publisher: In Easy Steps
Price: £10.99
ISBN: 978-1840786415
magpi.cc/eYrLzA

Whether you've moved sideways into project management, or you just need a quick refresher, this concise yet comprehensive guide will boost you through the jargon and well into getting the best out of the process.

SQL IN EASY STEPS



Author: Mike McGrath
Publisher: In Easy Steps
Price: £10.99
ISBN: 978-1840785432
magpi.cc/RfeNBw

If you want to master the basics of relational databases, or need an easy-to-use reference for fixing your website's database back end, this clear guide will take you from primary keys to outer joins.

HELLO SCRATCH!

Authors: Gabriel Ford, Sadie Ford, Melissa Ford
Publisher: Manning
Price: £18.99
ISBN: 978-1617294259
magpi.cc/DRWKuC

This book, like the colourful and easy-to-use Scratch programming language itself, is aimed at 8- to 14-year-olds. What's different is that it was written by twin 12-year-old coders, Gabriel and Sadie – with some writing assistance from their mother, Melissa Ford. We tell you this to emphasise that the writers know their audience, and not because any apology needs to be made for the book itself. This is a well structured and easy-to-follow introduction, with plenty to keep the reader interested.

First comes an introduction to the basics of the Scratch programming environment, then straight into creating art – for game sprites – then some basic coding concepts.

The eight most common blocks used in game building are explained, and you're ready to move onto the next section and code your first game.

Breakfast Wars, a two player ball-and-paddle game, is a fun reworking of Pong, taking the learner from co-ordinates to conditionals. Further games build your skills until the final platform game project introduces arrays and the effects of gravity. Lastly, a look at where to go next. A clear, useful, and enjoyable introduction to programming games in Scratch.

Score ★★★★★

THE PRODUCT IS DOCS

Authors: Christopher Gales, the Splunk Documentation Team
Publisher: Splunk
Price: £9.71
ISBN: 978-1973589402
magpi.cc/SZqlap

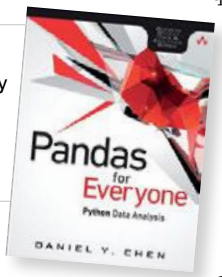
Quality of documentation is somewhat unevenly distributed around software projects. Many managers fail to prioritise documentation in fast-moving projects, and Agile development practices make it harder to fit in a documentation team with developers who "move fast and break things". However, there are projects that successfully deliver great documentation – Splunk is one of them, and here their documentation team outline some best practices for many parts of the process.

A collection of short essays, in alphabetically ordered chapters, this book can be dipped into for advice on Maintaining Existing Content, and Tools and Content Delivery, or read through from Agile to Working with User Experience and Design. Either way, you'll benefit from the experiences of a team who've learned to put the users first, and have a realistic view of what it's possible to achieve.

This is a practical book – so a chapter on Research for Technical Writers encourages a journalistic approach to tackling the right questions for the reader; the Learning Objective Matrix keeps documents at the service of user goals; Technical Verification is a brief but vital look at the occasionally neglected topic of reviews; and six of the chapters focus on working with others – from Customer Support to Product Management. This advice will improve any software product.

Score ★★★★½

PANDAS FOR EVERYONE



Author: Daniel Chen
Publisher: Addison-Wesley
Price: £29.99
ISBN: 978-0134546933
magpi.cc/Wryxyq

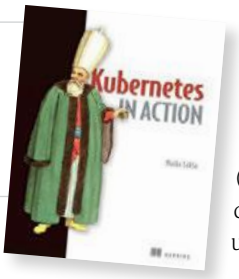
What is the ideal background for a data scientist? Possibly Chen's: the author came from a background in neuroscience and then public health, learnt R and data science, then learnt Pandas and the SciPy stack. A well-balanced immersion in the subject which is reflected in a comprehensive book with something to offer every level of Python programmer – beginners get a rapid intro to the language in the appendixes, which will underpin the main sections' practical learning.

Pandas is a tool for data manipulation and analysis which grew out of a project to perform quantitative analysis on financial data. It uses Series and DataFrame data types for tabular data (think spreadsheet or SQL table), and is a high-performance tool for loading, manipulating, aligning, and merging these data.

Chen dives straight into using a DataFrame object to work through the concept practically, then looks at plotting the data. Basics covered, it's time for a deep dive into 'tidy data', and plenty of cleaning and munging. A nice section on modelling data has some valuable tips for getting the best out of a data set, then the book rounds off with routes into the wider Python ecosystem.

Score ★★★★☆

KUBERNETES IN ACTION



Author: Marko Lukša
Publisher: Manning
Price: £40.99
ISBN: 978-1617293726
magpi.cc/Pydxlb

OS-level virtualisation has solved a lot of problems of scale – whether in the data centre, or in the development environment of your laptop – for distributing software over several containers, but brought with it the headache of managing it all. Kubernetes allows the orchestration of complex distributed apps, and its power and flexibility means that there's a lot to learn, whether you're a developer looking to lean on it heavily, or are tasked with managing the setup.

Kubernetes clusters containers in pods, and it's towards this fundamental structure that the opening chapters ascend after introducing you to your first Docker container. Next, setting up a Kubernetes cluster, then using automatic restarts to keep everything running healthily. Service discovery, shared persistent storage, config data, secret data (credentials), monitoring, and deployment follow, to get you up and running.

Practical walk-throughs are balanced with useful amounts of background information, and insights into real-world use. Building on the practical first half, this progressive introduction then tackles the higher-level concepts, feeding back this new understanding into better deployments. Excellent diagrams, and layout and ordering of the information, offer a clear understanding of an essential but rather complex tool.

Score ★★★★★

ESSENTIAL READING: C# ESSENTIALS

Microsoft's Java-ish mainstay of the .NET platform is open source, and runs on your Pi.

C# Programming Fundamentals: Lesson 1 - Using Data

Author: Jesse Dietrichson
Publisher: YouTube
Price: Free video
magpi.cc/EkwwMf

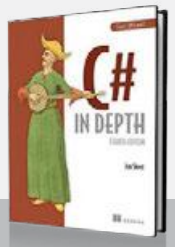
Well-regarded series of YouTube vids that will give you real understanding of far more than just the basics.

Gray Hat C#

Author: Brandon Perry
Publisher: No Starch
Price: £31.99
ISBN: 978-1593277598
magpi.cc/nlApKt

Excellent whirlwind language introduction, followed by a great set of automation examples and some handy pen-testing material. Recommended.

C# in Depth, 4E



Author: Jon Skeet
Publisher: Manning
Price: £27.99
ISBN: 978-1617294532
magpi.cc/hTZjAg

Go beyond the basics, with an excellent dive into generics, delegates, lambda expressions, LINQ, and more.

Functional Programming in C#

Author: Enrico Buonanno
Publisher: Manning
Price: £27.99
ISBN: 978-1617293955
magpi.cc/rPSZdo

Get inside the functional programming mindset and do more with less code. This will change the way you program.

More Effective C#

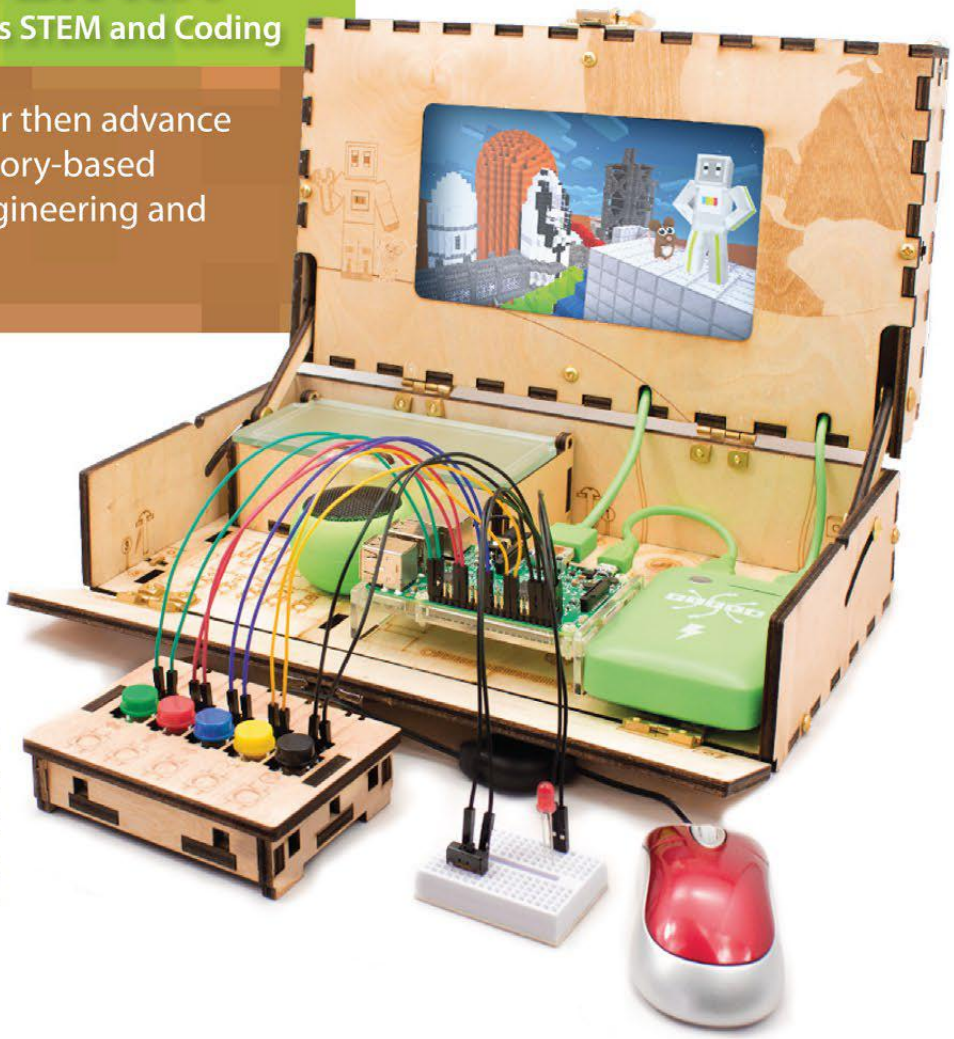
Author: Bill Wagner
Publisher: Addison Wesley
Price: £32.99
ISBN: 978-0672337888
magpi.cc/Irmnok

Thoroughly readable and very practical guide to some advanced-level C# topics from a member of the ECMA C# Standards Committee.

PIPER COMPUTER KIT

Educational Computer that teaches STEM and Coding

Kids build their first real computer then advance through Piper's award-winning story-based curriculum and learn physical engineering and electronics in the process.



Special $10 MagPi coupon:

MagPiSummer

only at BuildPiper.com

"It's a really fun educational computer kit that should really impress those who love Minecraft and building stuff" — The MagPi

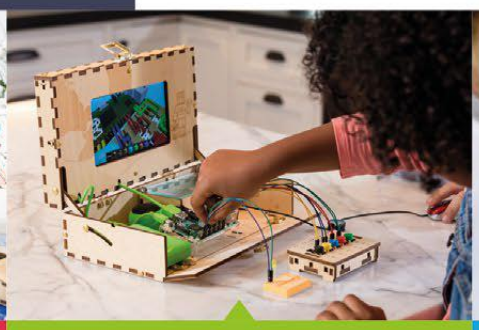
A STEM SOLUTION FOR SCHOOLS

Teachers all over the world use Piper to inspire kids to program, design, and engineer.

Want to learn how to bring Piper to your school?
buildpiper.com/EDU

THE PIPER EXPERIENCE

BUILD FROM SCRATCH



SOFTWARE-BASED CURRICULUM

CREATE YOUR ELECTRONICS

Supported by Top University Funds
Stanford-StartX Fund
AEF of Princeton University

Available at:

PIPER
BuildPiper.com

amazon

ToysRUs

BARNES&NOBLE

MONITORING A LAB WITH RASPBERRY PI

How are students using a university lab? One lecturer has a way to know

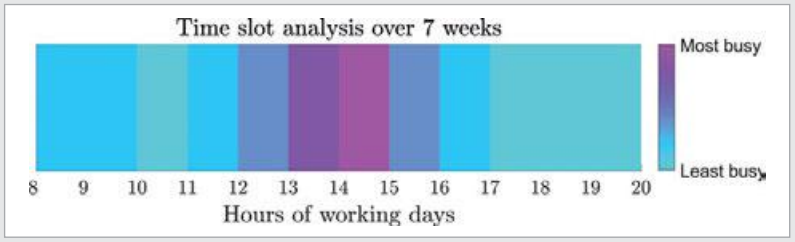
Improving a user experience is an important part of many operations, especially when it's students trying to use a lab for their university work. Instead of giving students endless surveys, English phonetics lecturer Emmanuel Ferragne decided to use the Raspberry Pi to track how the labs were being used, in an effort to make them better.

Emmanuel

Nmae: Emmanuel Ferragne
Occupation: University Lecturer

What are you using the Raspberry Pi for?
I call it the Lab Monitor for want of a better label. Basically I had all those expensive, state-of-the-art instruments from a number of research projects, but they were stored in a cupboard because no room was available. Things have changed very recently and I now have a small lab where my students can use the hardware for their own research. Rather than imposing strict rules, I thought I would collect data to understand how students actually use the lab. So the Lab Monitor is really there to learn from the students' habits and ensure that the lab meets their needs, and not the other way around!

I must insist that this is not about surveillance at all. I have no means of knowing who exactly is using the lab at any given time – and I don't want to know! If the Lab Monitor tells me there's activity going on in the lab, then I'm more than happy.



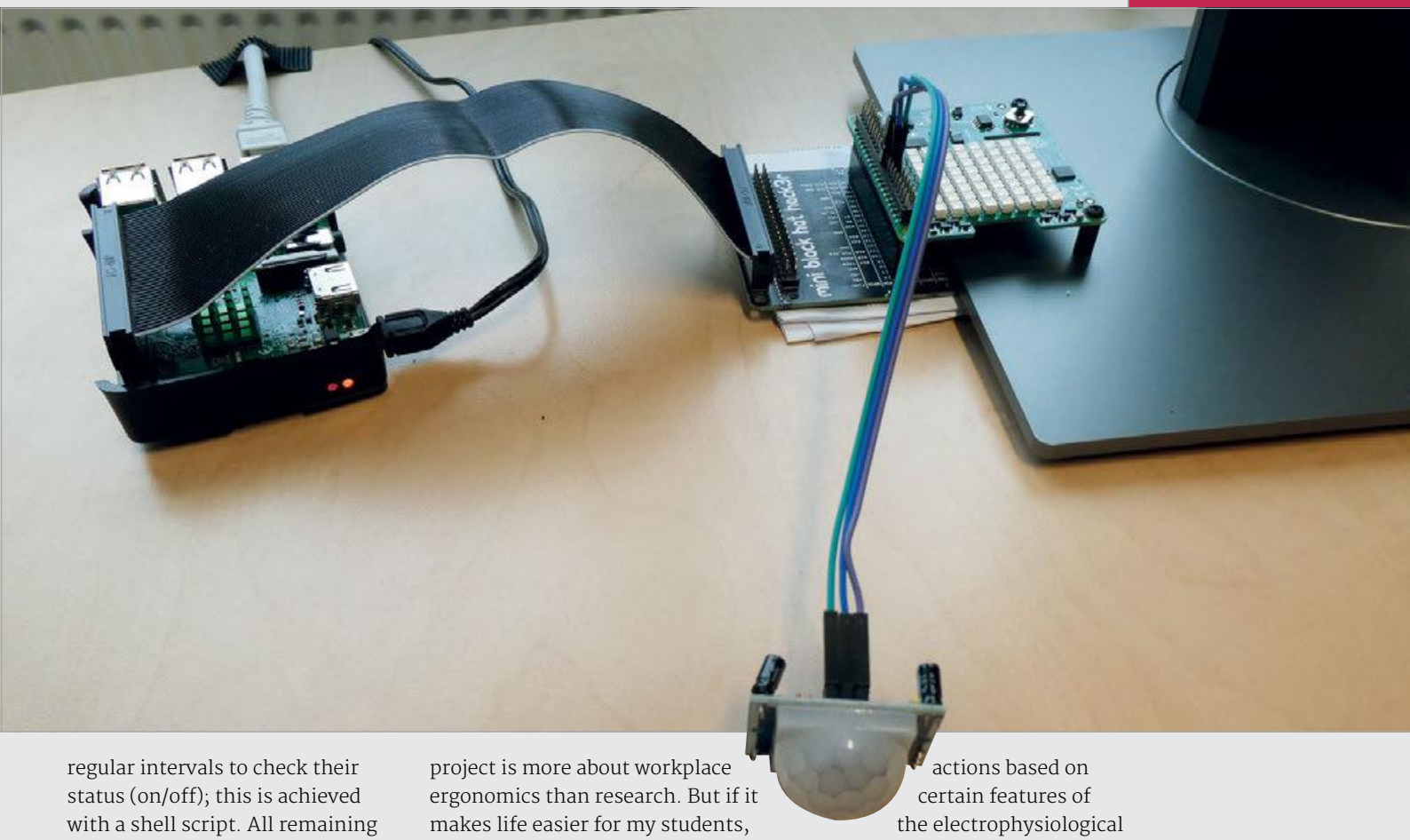
Above With the data, the lab can be improved and streamlined for students

What kind of research are you doing?
I'm involved in a number of projects related to the acquisition of English by French learners. So I use techniques like acoustic speech processing, ultrasound tongue imaging, electroencephalography (EEG), etc., to better understand how learners produce and perceive the sounds of a foreign language. More generally, I'm interested in all aspects of the human voice: what it says about you, how it changes depending on who you are talking to, to what extent somebody's voice can be said to be unique, etc.

Why the Raspberry Pi?
I'm always on the lookout for new hardware, and over the years I've developed a keen interest in tweaking the instruments I use for my research. One problem with the professional-grade instruments I work with is that some of them come with limited hackability, which I find ever so frustrating! But I knew the Raspberry Pi would let me be creative.

How does the system work?
In the current version of the Lab Monitor, on the hardware side, there's a Raspberry Pi 3, a Sense HAT, an infrared motion detector, and a Mini Black HAT Hack3r. The Sense HAT gathers temperature and humidity data, and students can move the joystick if they don't want to be disturbed when they run experiments. The Sense HAT is connected to the Pi via a Black HAT Hack3r so as to obtain more accurate temperature readings.

On the software side, the Nmap program scans the IP addresses of the computers in the lab at



regular intervals to check their status (on/off); this is achieved with a shell script. All remaining operations are managed by a Simulink (a visual programming language by MathWorks) model that I deployed to the Pi. It collects the output from the IP scan, the temperature, humidity, joystick, and motion sensors, and sends the data to ThingSpeak, which is the MathWorks IoT platform. The IoT channel is public, so all users can check if, for example, the computer that is connected to the special instrument they are planning to use is available.

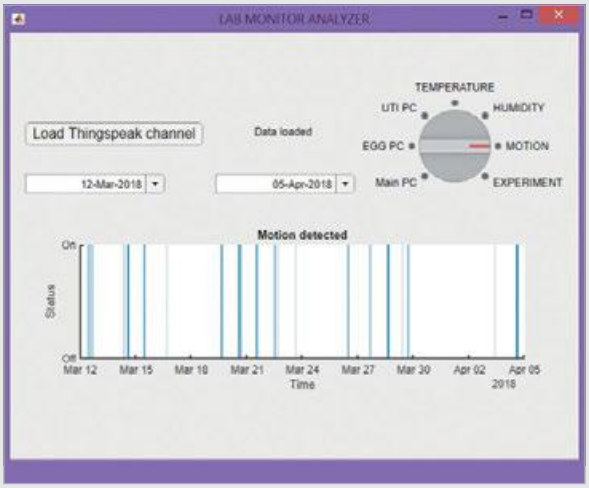
I have developed a Matlab GUI that lets me import the data feed and analyse it offline. For instance, I can check if some computers are over-/under-used and modify the lab setup accordingly. Or I can identify empty time slots to plan new lab meetings. Temperatures in the building can go pretty high in the summertime so I'll keep an eye on them too.

Do you have any future plans for the research setup?

Well, strictly speaking, this project is more about workplace ergonomics than research. But if it makes life easier for my students, then it will have a positive impact on their research. And yes, I have plans to extend the system to other rooms here at the university, provided I get money to buy the hardware. But before that, two aspects should be improved. First, some downsizing is in order: for example, the Sense HAT here is overkill, and also, I might consider using a Pi Zero instead. And second, I have yet to find or build a suitable enclosure.

Do you have any future plans for using the Raspberry Pi in other research?

Yes, among other things, I sometimes work with a wonderful, tiny piece of hardware for biosignal acquisition called BITalino (**bitalino.com**). In our current setup, the BITalino gathers data and sends it to a PC over Bluetooth. If we replace the PC with a Pi, the system becomes totally wearable and we can run our experiments anywhere. And the Pi could process the data and trigger actions based on certain features of the electrophysiological signal. In an experiment we are carrying out right now, we're recording people's heart rate and electrodermal activity in response to emotionally loaded words. It is easy to imagine how we could program the Pi to wait for these two signals to return to some 'resting state' before playing back the next audio stimulus. The Pi could also sound a buzzer if a participant's heart rate goes above or below certain thresholds.



Below Anonymous data is used to analyse how the lab has been used

THE MONTH IN RASPBERRY PI

Everything else that happened this month in the world of Raspberry Pi

MAGPI MONDAY

DO YOU GET UP TO ANY FUN RASPBERRY PI PROJECTS AT THE WEEKEND? SHOW US ON TWITTER!

Every Monday we like to ask our readers what they got up to with their Raspberry Pi boards over the weekend. A lot of you seem to do some fun stuff, and we're always happy to see what people have made! Send us your projects on a Monday morning to **twitter.com/TheMagP1**.

Mark Cantrill
@AstroDesignsLtd
Following
Replying to @TheMagP1
The #MatchBot got a makeover this w/e - slightly neater, simpler, more reproducible, a little bigger but still some room for improvement...
Mark Cantrill @AstroDesignsLtd
Continuous rotation servos = no need for a motor driver! Just a #RaspberryPi one battery and two continuous rotation servos. Need to make it even smaller next time...
0:18
11:18 PM - 9 Apr 2018

UPDATED MATCHBOT

A classic Pi Zero project from when the miniature Raspberry Pi was launched in 2015, the original has received a major (minor?) upgrade.

POOPICORN

An interactive donation counter for a new creative play space in Leeds, featuring the Ratvaark.

TINYPI

It's always fun to see what the TinyPi is up to – we like the Game Boy design here!

BIRDBOX

We love seeing nature projects, especially ones that have a bit of nature in action!

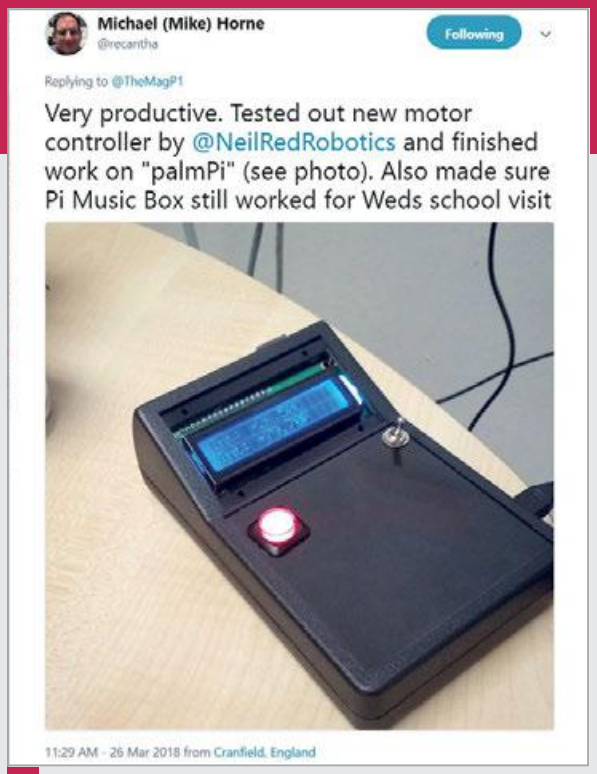
BERRYCADE

This project has since been completed by Les. The Berrycade is a great-looking custom arcade build!

biglesp
@biglesp
Following
Replying to @TheMagPi
I've got two. My #berrycade project is almost finished and I saw my latest Pi Zero W Weather Station kit IRL!
Weather Station
10:30 AM - 26 Mar 2018

MUSIC PLAYOUT GUI

For aspiring DJs and theatre techs, this system lets you control music output via a Raspberry Pi.


Michael (Mike) Horne
@recantha
Following
Replying to @TheMagPi
Very productive. Tested out new motor controller by @NeilRedRobotics and finished work on "palmPi" (see photo). Also made sure Pi Music Box still worked for Weds school visit
11:29 AM - 26 Mar 2018 from Cranfield, England

PALMPI

A neat-looking handheld Raspberry Pi device made by CamJam/Pi Wars organiser Michael Horne. It records data from sensors to read on the screen.

A MONTH IN RASPBERRY JAMS

Jams are back to business as usual now the Birthday Weekend is over

We always love seeing the amazing pictures from Raspberry Jams around the world, and April was no different! Whether it was on a weekend, holiday, or a normal Tuesday, Raspberry Pi fans getting together to have fun with their Pi is always great to see.

Kyle Wilke
@KyleJwilke
Follow
So much going on at the @SVUSDSchools #SaddlebackValleyJam @TheMagP1 reading area. @Sonic_Pi station using old PCs with Raspian Pixel. 8 @Raspberry_Pi stations for hands on workshops! What a jam!
3:00 AM - 25 Mar 2018
Saddleback Valley, CA, USA

Preston Raspberry Jam
@PrestonRJam
Follow
Photos from last night's Preston Raspberry Jam. Our next event is on Monday 7th May (Bank Holiday).
Preston, UK

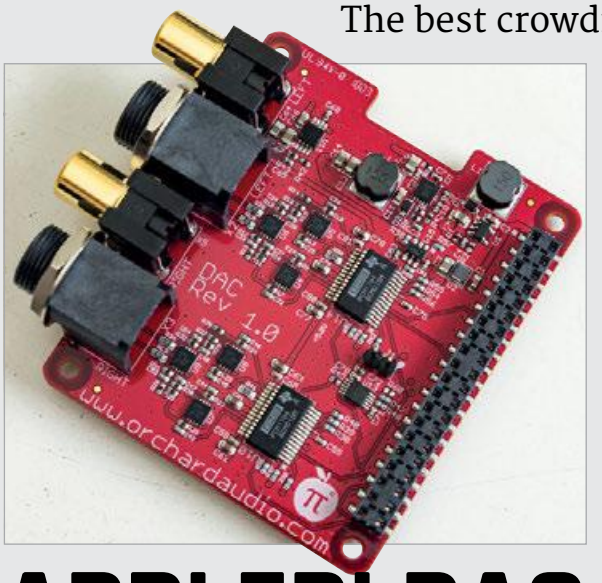
NI Raspberry Jam
@NIRaspJam
Follow
We are busy at today's NI @Raspberry_Pi Jam at @QUBMathsPhys with workshops from unicorn horn hacking (with @adafruit Circuit Playground Expresses) to @4tronix_uk Bit:Bots to Sense HATs!
Plus, some amazing cupcakes from @Baker_Geek!
3:07 PM - 24 Mar 2018
Northern Ireland, UK

Gabriel Garcia
@GabrielGarciaR
Follow
Haciendo presencia hoy en el Arduino Day 2018 & Raspberry Jam Big Birthday Weekend. @iotcolombia @arduino @ArduinoBogota @raspberrypi_co @Raspberry_Pi #ArduinoDay2018 #RaspberryPi @MakersColombia
Bogotá, Colombia

York, UK

CROWDFUND THIS!

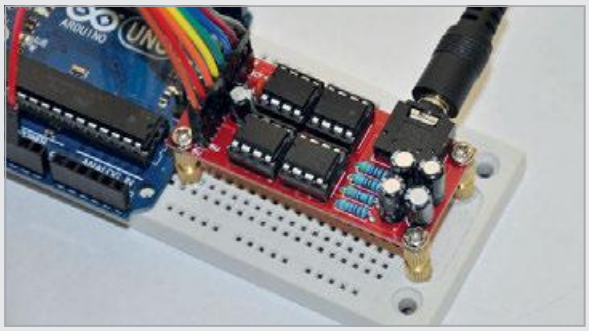
The best crowdfunding hits this month for you to check out...



APPLEPI DAC

kck.st/2Heej1F

The ApplePi DAC claims to be the highest-quality DAC that you can get for the Raspberry Pi. "For less than $250 [£176] (ApplePi + Raspberry Pi), you can achieve performance currently only available at $750+ [£523]," says the creator Leo Ayzenshtat on the Kickstarter page. The device will retail at $200 (£141) when launched, but you can get one as a reward for $175 (£123).



THE BIG BUDDY TALKER

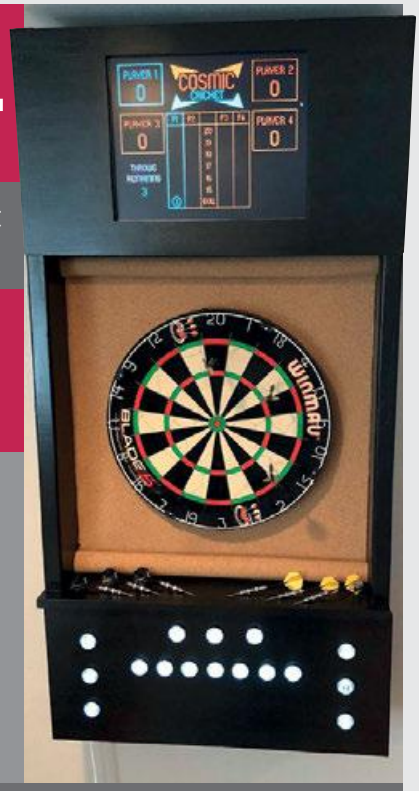
kck.st/2pxiKtz

The Big Buddy Talker is a way to add speech to projects that use the Raspberry Pi. It's a physical board that contains over 1000 words that it can speak in a pretty natural voice. A total of 500 backers will get to add a word to the library and at the time of writing there have been just shy of 300 backers. So if you want to add your favourite swear-word... you might need help, but you can also request it if you pledge.

BEST OF THE REST

Here are some other great things we saw this month

RASPBERRY PI DARTBOARD SCORER



We're seeing a few Pi-powered score-boards recently, but we quite like this dartboard cabinet with built-in score-board. You do need to use the physical buttons on the cabinet, though: it's not smart enough to automatically sense where the darts have landed.

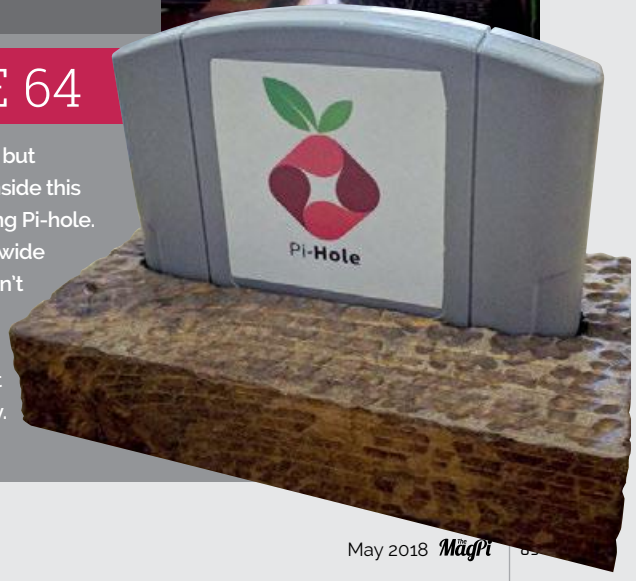
magpi.cc/ZSKYuz

DIY SELFIE STICK

"Made a selfie stick. Am I trending now?" posted RulerOfTheThirdRealm on the Raspberry Pi subreddit. Well, you may not be trending for this wonderfully DIY selfie stick, but at least you're in *The MagPi*.

magpi.cc/WazQBN

PI-HOLE 64



It may not look like it but there's a Pi Zero W inside this N64 cartridge, running Pi-hole. Pi-hole is a network-wide ad blocker, so we don't know why it's been put inside an N64 cartridge, although it does look very pretty.

magpi.cc/yrbuXh

COMMUNITY PROFILE

CICELY DAY

California-based robot lady with the magic touch

Cicely

Category: Educator

Day job: Teacher and coding club leader

Website: magpi.cc/LnYTmU

twitter.com/cutenose76

Cicely Day has an incredibly infectious spirit that brightens any room. She is genuinely loved by her students who are always delighted to come to the after-school coding clubs she leads on Mondays and Tuesdays at Burckhalter Elementary School in Oakland, California.

On Mondays and Tuesdays, students explore a wide range of activities: Sense HAT animations, creating stories in Scratch, physical computing with Raspberry Pi, and homemade toy hack projects. When it comes to her students' learning, Cicely prefers to take a hands-off approach and let them lead their own journey. "I don't want to be a sage on the stage. I try to stay out of the way as much as I can because the more I'm in the way… the less they can problem-solve, figure it out for themselves, and collaborate together."

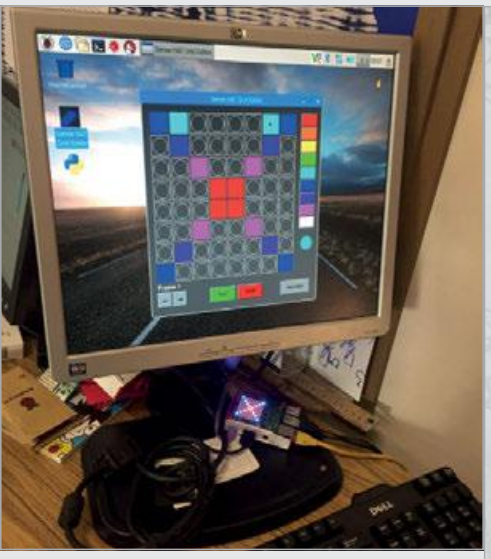
Certified Educator

Cicely attended the very first Picademy at the Computer History Museum in 2016. As a Raspberry Pi Certified Educator, she frequently connects with other educators through the hashtags #picademy and #pichatusa on Twitter, and at educational conferences such as SXSW EDU. This network has been very helpful for inspiration, tips, and advice when she is looking for engaging projects to share with her students. When Cicely first purchased her Raspberry Pi, she wasn't certain how to turn it on, she didn't have a monitor, and wasn't sure how the pieces would all fit together. But, having such a strong support system from

HIGHLIGHT

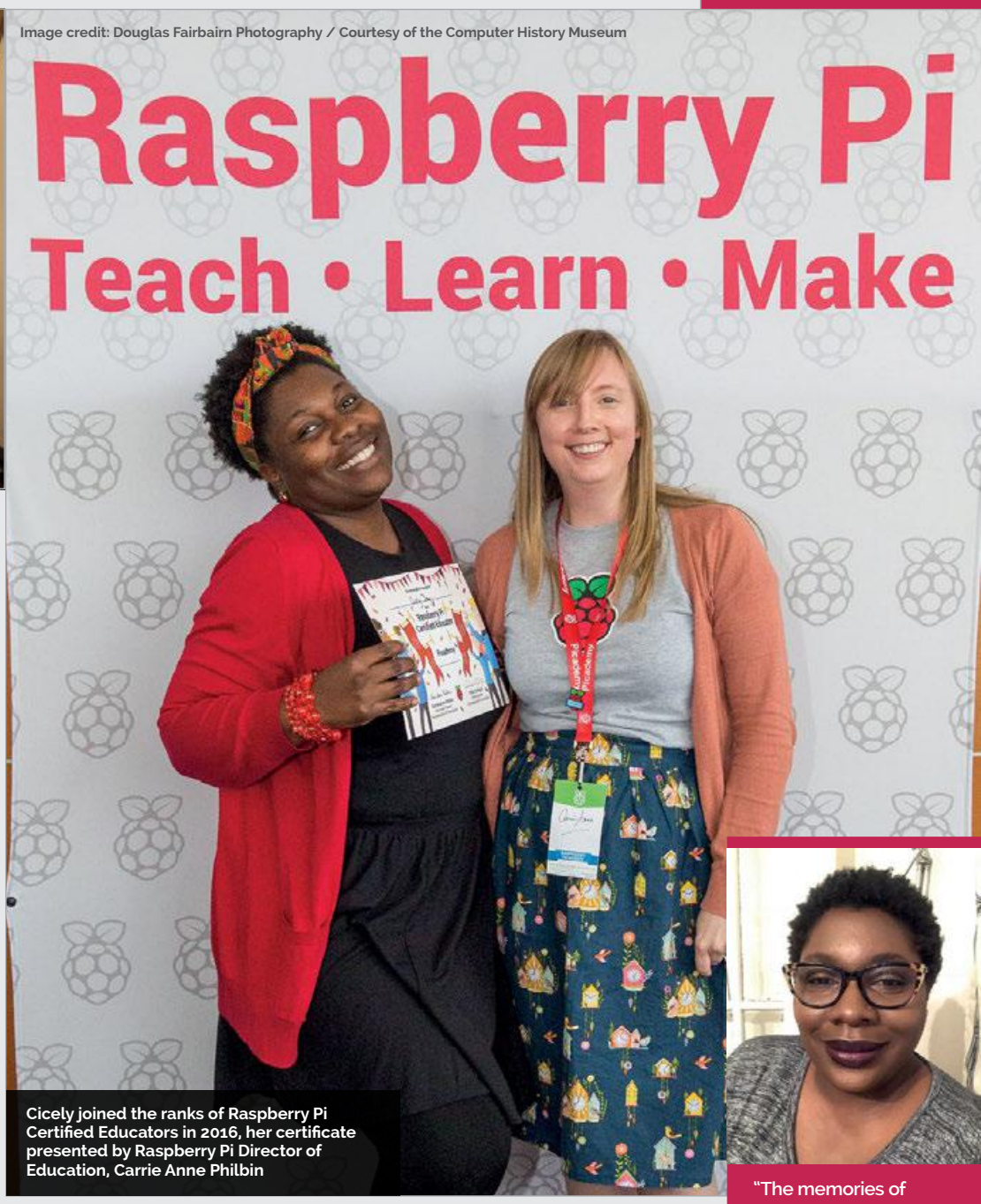
EXPLORATION IN THE CLASSROOM

To keep her students engaged, Cicely experiments between mixing up different types of activities and sometimes just going with the flow. Some days she leads her students through hacking with Minecraft and on other days they are free to explore any creative project with their friends. Her classroom has become a space for all children, regardless of their background, to be creative, explore new activities, make mistakes, and learn collaboratively with each other.



Above Cicely rigged up the Sense HAT grid editor for her club, controlling physical LEDs via a desktop GUI (magpi.cc/2dXmIbn)

Image credit: Douglas Fairbairn Photography / Courtesy of the Computer History Museum



Cicely joined the ranks of Raspberry Pi Certified Educators in 2016, her certificate presented by Raspberry Pi Director of Education, Carrie Anne Philbin

the Raspberry Pi community and her fellow educators has allowed Cicely to challenge herself and to continue her learning. "I'm a Raspberry Certified Educator, I have to do it, I have to try, they believe in me."

At a recent Coding Club, Cicely asked one of her students, Jania, how digital making, coding, and computer science have impacted her learning. Cicely was touched by Jania's reply. Jania's grandmother used to sit with her and show her how to use the keyboard and the computer. Since her grandmother passed away, Jania wanted to learn more about computers and feels that if she becomes a computer scientist and a programmer, she can change the world and can make things better.

Opportunities for all

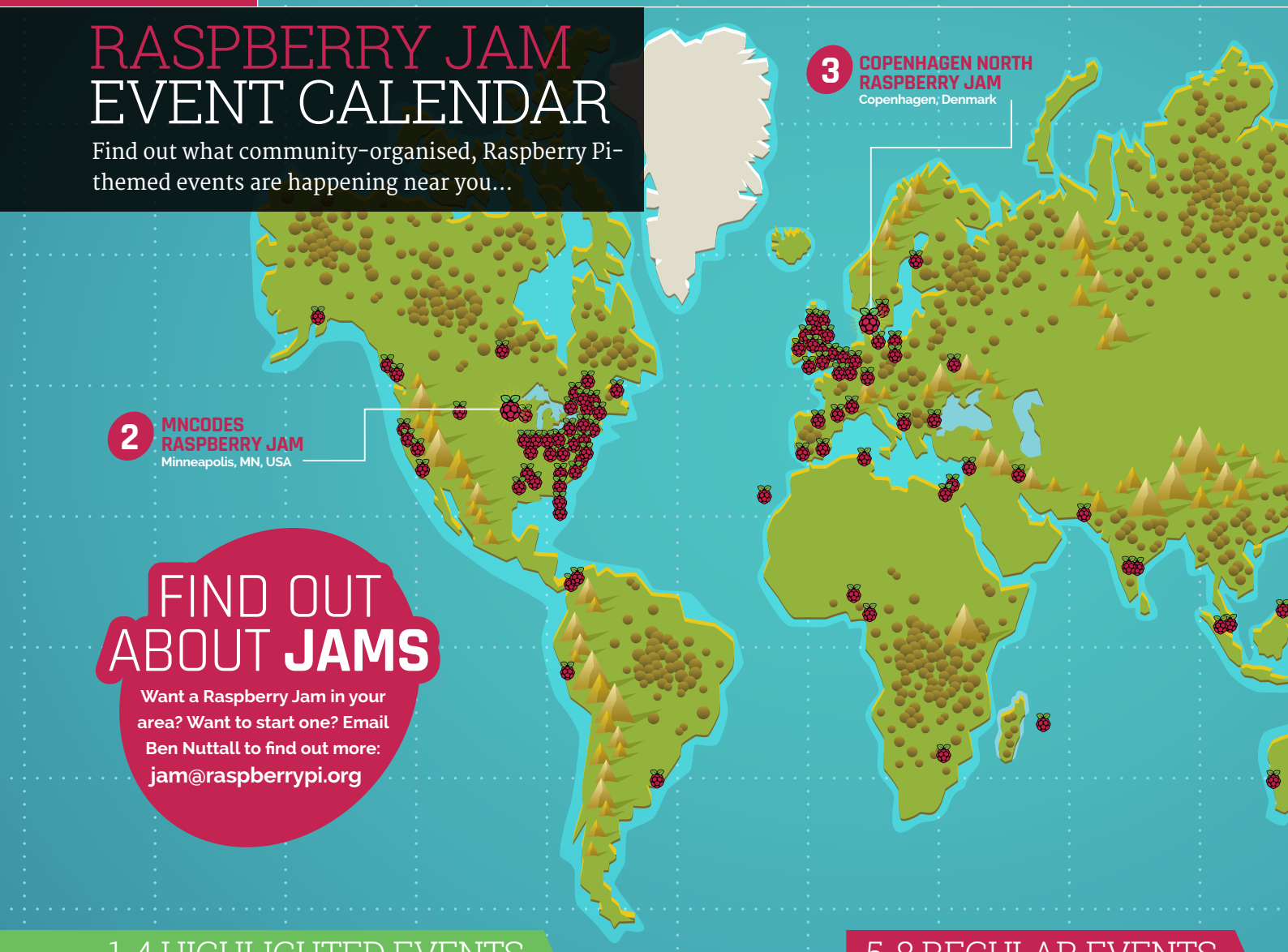
Cicely introduces computer science to young kids in a way that allows them to explore their creative side while showing them options and pathways that they have never dreamed would be possible. "That's my mission in life. I have been able to become a teacher, with the ability and access to be with children who would otherwise get written off because they look like me. I want all of our students who are brown, minorities, and are from different places to know that there are people who look like them and are geeky and like to make cool projects with technology. I can't say that everyone who comes through my classroom is going to be a computer scientist. However, I want to give them that access and choice. I don't want racism, stereotypes, prejudice, and institutionalised foolishness to make that choice for them. I want to empower them to figure out to how they fit into this world and to figure out their way. I'm just the lady that has the robots and they are the ones that do all the work. I just say hey, have fun and if you're stuck, let me know…"

> "I have to do it, I have to try, they believe in me"

"The memories of my mother as a woman of many resources and a jack-of-all-trades has inspired me to keep the tradition going," Cicely explains in her blog, going on to attribute her desire to make things to her mother's life lessons in problem solving, working with others, and critical thinking.

RASPBERRY JAM EVENT CALENDAR

Find out what community-organised, Raspberry Pi-themed events are happening near you...



FIND OUT ABOUT **JAMS**

Want a Raspberry Jam in your area? Want to start one? Email Ben Nuttall to find out more: jam@raspberrypi.org

1-4 HIGHLIGHTED EVENTS

1 THORPE RASPBERRY JAM

When: Saturday 5 May
Where: Thorpe St Andrew High School & Sixth Form, Thorpe St Andrew, UK
magpi.cc/DAQVyt
A community meet-up for people to come together to learn about digital making with the Raspberry Pi.

2 MNCODES RASPBERRY JAM

When: Monday 14 May
Where: University of Minnesota, Minneapolis, MN, USA
magpi.cc/rotiTV
A place to learn, share, and socialise in a hands-on digital making community focused on the Raspberry Pi.

3 COPENHAGEN NORTH RASPBERRY JAM

When: Thursday 17 May
Where: Nærum Gymnasium, Copenhagen, Denmark
magpi.cc/eNdjCB
To celebrate the students from Nærum Gymnasium being chosen as Astro Pi winners, a Raspberry Jam is being held!

4 JAMMING IN MARLBOROUGH

When: Sunday 20 May
Where: Marlborough Town Hall, Marlborough, UK
magpi.cc/KiqImM
Introducing fun and educational digital making activities to both adults and young people.

5-8 REGULAR EVENTS

5 HULL RASPBERRY JAM

When: Saturday 5 May
Where: Hull Central Library, Hull, UK
magpi.cc/pTwcMJ
This month is a robot spectacular! Learn all about hobby robotics from Pi Wars contestants.

6 PRESTON RASPBERRY JAM

When: Monday 7 May
Where: Media Factory Building, Preston, UK
magpi.cc/LjekqU
A community of people who meet each month to learn, create, and share the potential of the Raspberry Pi.

WE'VE HIGHLIGHTED SOME OF THE AREAS IN NEED OF A JAM! CAN YOU HELP OUT?
5 HULL RASPBERRY JAM
Hull, UK
1 THORPE RASPBERRY JAM
Thorpe St Andrew, UK
6 PRESTON RASPBERRY JAM
Preston, UK
8 MANCHESTER RASPBERRY JAM
Manchester, UK
4 JAMMING IN MARLBOROUGH
Marlborough, UK
7 CORNWALL TECH JAM
Bodmin, UK

7 CORNWALL TECH JAM

When: Saturday 12 May
Where: Chy Trevail, Beacon Technology Park, Bodmin, UK
cornwalltechjam.uk
Learn about programming on a variety of platforms, including Arduino and Raspberry Pi, in various languages.

8 MANCHESTER RASPBERRY JAM

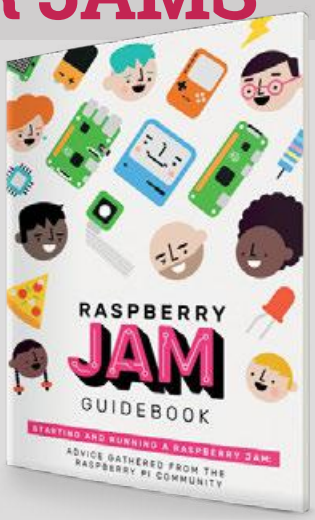
When: Saturday 12 May
Where: The Shed, Manchester, UK
magpi.cc/PpKycI
Take part in workshops taught by experts or get help working on your own Raspberry Pi projects.

RASPBERRY JAM ADVICE

FUNDING YOUR JAMS

> "We have some SD cards which we loan out for the afternoon, and other cards which we sell at a small profit. If you can find a cheap source of decent blank SD cards, then it's a good way to raise funds while also solving a big problem."
>
> **Andrew Oakley**
> **Cotswold Jam**

Every Raspberry Jam is entitled to apply for a Jam starter kit, which includes magazine issues, printed worksheets, stickers, flyers, and more. Get the guidebook here: **magpi.cc/2q9DHfQ**



YOUR LETTERS

COLD CLUB

Greetings from Norway. Colder than ever. I am having a weekly code club in Sandefjord, Norway, for kids from eleven and up (**magpi.cc/oLiYvu**). We are doing a lot of things, from coding to building an arcade running on Raspberry Pi and RetroPie. Yesterday we put together the cardboard Google AIY Voice Kit.

In May we have been invited to a retro-game convention in our town. And we want to bring stuff to show kids about being creative.

The MagPi is a very good place to start for a lot of things, and I wonder if you may be interested in sponsoring us with something we can showcase, and then use in our club later on?

We already have some Pi Zeros we want to develop with, and I also want to make robots. A Norwegian site sponsored us with two robots for the convention.

Last year people wanted information, and we want to be informative.

Terje Rove Pettersen

Unfortunately, we are unable to help out with any sponsorships ourselves, but if there are any kind readers in Norway who have some spare Raspberry Pi kit, hopefully they'll read this and get in touch with you via your Facebook page: **magpi.cc/oLiYvu**.

Do remember though that all issues of our magazine, including our Projects books and Essentials books, are available for free as PDFs. You can give those out for free to participants to help continue their learning.

KERNEL OF AN IDEA

I was hoping to find an article about the new kernel (4.14) in issue 68, but now I only hope I'll see one on the Raspberry Pi blog. I asked about this in the forums and I got an answer, but I wanted more specific info if possible. There are a lot of small updates such as 'improved Raspberry Pi 3 support' and 'new drivers'.

What do each of these updates mean? I'm sure I wouldn't be the only one who would love to know more.

Nicolae Crefelean

As we're less of a developer magazine and more of a community one, getting into the nitty-gritty about kernel updates is usually a bit lower in priority when other stories are competing for magazine space.

As for getting the specific answers on the blog, usually a lot of the vague updates on these kernel patch notes are minor performance improvements that will barely be noticed. If there's anything major, though, one of the Raspberry Pi engineers will write something about it!

RETRO COMPUTING: RESPONSE!

I have just finished reading Dexter N Muir's letter in *The MagPi* issue 68, April 2018, and I might have good news for him regarding an up-to-date emulator.

There are still a few around. UQLx is one, available from **github.com/SinclairQL**, but it is source code only and will need compiling on the Raspberry Pi with a plain C compiler such as GCC. The binaries available at **magpi.cc/GMihLY** are for ARM 6, ARM 7, ARM 8, and 64-bit PCs. I think I might even have the 32-bit version – compiled for the original Raspberry Pi Model B, circa 2012, with 256MB RAM – lying around somewhere. If Dexter wants a copy, and I still have it, please pass on my email address to him.

There is a very useful page about QL emulators at **magpi.cc/GmxuSG**.

You can find SMSQmulator (no 'e') here: **magpi.cc/yiKBtb**. This emulates an 'ideal' QL, as opposed to the actual QL. This emulator is regularly updated by the author Wolfgang Lenerz. It's runnable under Java Runtime versions 7 and 8. Later Java versions might also work.

There is still, after all these years, an active forum at **qlforum.co.uk**; a newly converted wiki at **qlwiki.qlforum.co.uk**; and lots of free software, documentation, etc. at the Sinclair QL Homepage, run by Dilwyn Jones, at **dilwyn.me.uk**.

Finally, there's an active QL mailing list which can be subscribed to at **magpi.cc/grokxY**.

Norman Dunbar

Thank you very much for providing this information! The emulation community is forever changing, which can make it slightly tricky for us outsiders to keep up with unless we're actually in the community!

Hopefully Dexter, and others, will find this information useful for future emulation efforts. We certainly think it's great!

For more information about retro computing on your Raspberry Pi, take a look at issue 67!

FROM THE FORUM:

SENDING PROJECTS TO THE MAGPI

The Raspberry Pi Forum is a hotbed of conversations and problem-solving for the community – join in via **raspberrypi.org/forums**

What would be the current, correct channels to suggest topics for an article or two in future issues of *The MagPi*?

Alternately, where would one send article drafts, if there is the possibility of community writers?

There appear to be several email addresses listed over the years for that purpose, and the one from a more recent issue did not reply.

Typo Kingg

The best way to let us know topic suggestions, and if you want to pitch an article, is to send us a message directly. We definitely prefer if you email us (**magpi@raspberrypi.org**), but if you want to send us a message via Facebook (**facebook.com/MagPiMagazine**) or tweet to **@TheMagP1** with pictures and info, that's a great first step!

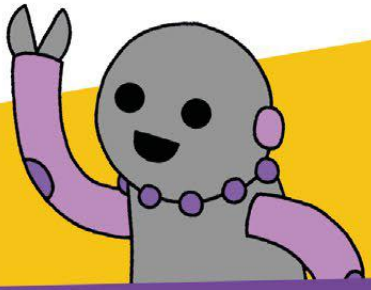
We always want to hear about what cool things you're doing with the Raspberry Pi.

WRITE TO US

Have you got something you'd like to say?
Get in touch via **magpi@raspberrypi.org** or on The MagPi section of the forum at: **raspberrypi.org/forums**

DINrPlate™
The simple way to mount your Pi!
• Industrial DIN rail mount
• Open frame for better airflow
• Integrated USB strain relief
www.DINrPlate.com

{code club}



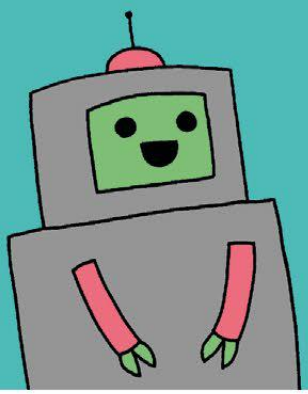
START A CODE CLUB IN YOUR SCHOOL!

Code Club is a network of volunteers and educators who run free coding clubs for young people aged 9-13.

Our aim is to inspire the next generation to get excited about computer science and digital making.

"We use Code Club's fun educational resources to run a weekly after-school club for Year 7 and Year 8 pupils. The students benefit considerably from the extra challenge!"

Karen Dadd, Computing Teacher



- **Code Club is free**
- **Code Club provides step-by-step guides for Scratch, Python, HTML, and Sonic Pi**
- **Code Club helps children develop skills including logical thinking, creativity, and resilience**

We have over 6000 clubs across the UK teaching more than 80,000 young people to code—come and join us!

Find out more at www.codeclub.org.uk

Code Club is part of the Raspberry Pi Foundation. Registered Charity Number 1129409

WIN!

Raspberry Pi

10 RASPBERRY PI 3B+ & PoE HATS

To celebrate the launch of the Raspberry Pi 3B+ and the new Power over Ethernet (PoE) accessory, we've got ten of each up for grabs...

The new Raspberry Pi 3B+ has a faster CPU, Ethernet, and wireless networking. Read all about it in issue 68 (**magpi.cc/68**).

The Raspberry Pi PoE HAT is a new accessory from Raspberry Pi. It is used to power a Raspberry Pi via an Ethernet cable. With the PoE HAT, you can attach a Raspberry Pi to a network and power it directly.

The PoE HAT will be available soon, and we have ten to give away to our readers (along with ten new Raspberry Pi 3B+ boards). Enter today for a chance to win.

Enter now at magpi.cc/win

Learn more:
magpi.cc/jLqtqL

Terms & Conditions

Competition opens on **25 April 2018** and closes on **31 May 2018**. Prize is offered to participants worldwide aged 13 or over, except employees of the Raspberry Pi Foundation, the prize supplier, their families or friends. Winners will be notified by email no more than 30 days after the competition closes. By entering the competition, the winner consents to any publicity generated from the competition, in print and online. Participants agree to receive occasional newsletters from The MagPi magazine. We don't like spam: participants' details will remain strictly confidential and won't be shared with third parties. Prizes are non-negotiable and no cash alternative will be offered. This promotion is in no way sponsored, endorsed or administered by, or associated with, Instagram or Facebook.

MATT RICHARDSON

Matt Richardson is the Executive Director of the Raspberry Pi Foundation North America and author of *Getting Started with Raspberry Pi*. Contact him on Twitter @**MattRichardson**.

INTERSECTING INTERESTS

While blending making and running, **Matt Richardson** finds more motivation for both

Those of us who read *The MagPi* regularly have something in common: we love the Raspberry Pi. And most of us like to use technology such as Raspberry Pi to create things, whether it's as a side hobby, for solving problems, or for professional projects. But for each of us, there is more to our lives; we also have interests such as music making, beer brewing, cycling, hiking, or knitting. The best Raspberry Pi projects involve other hobbies, disciplines, or fields.

I recently picked up long-distance running again as a New Year's resolution. I'm training for a half marathon and I hope to keep up my weekly mileage after my race. My problem is that I travel so often, it's hard to keep my training on track. To help motivate me, I'm using a Raspberry Pi to create a display board for digital running badges: one for each state and country that I log a run in.

> "I love projects that display a person's passion for technology and how their interests intersect"

Run for life

When I run, I track my myself with an app called Strava. I therefore started this project by using Python to read the Strava API. I then used the Google Maps API to take the longitude and latitude of the run and fetch the state and country where it took place. That information is stored in a simple database and can be refreshed on any interval. Using Flask, a great dynamic web framework for Python, I generate a webpage which shows each of the unique places I've run. The project has a long way to go, however. For example, it just shows a list of places, and there aren't actual graphical badges yet, but those will come soon, I hope.

The cool thing is that getting back into running motivated me to make this project and working on this project motivates me to run. Blending my interests of technology and running created a perfect positive feedback loop for me. For this project, I've learned a lot about working with different APIs and about using a database with Python. I've also stuck to my training plan and feel ready to run my half marathon next week. I'm confident that after the race, I'll be eager to keep earning badges for new running locations as I travel.

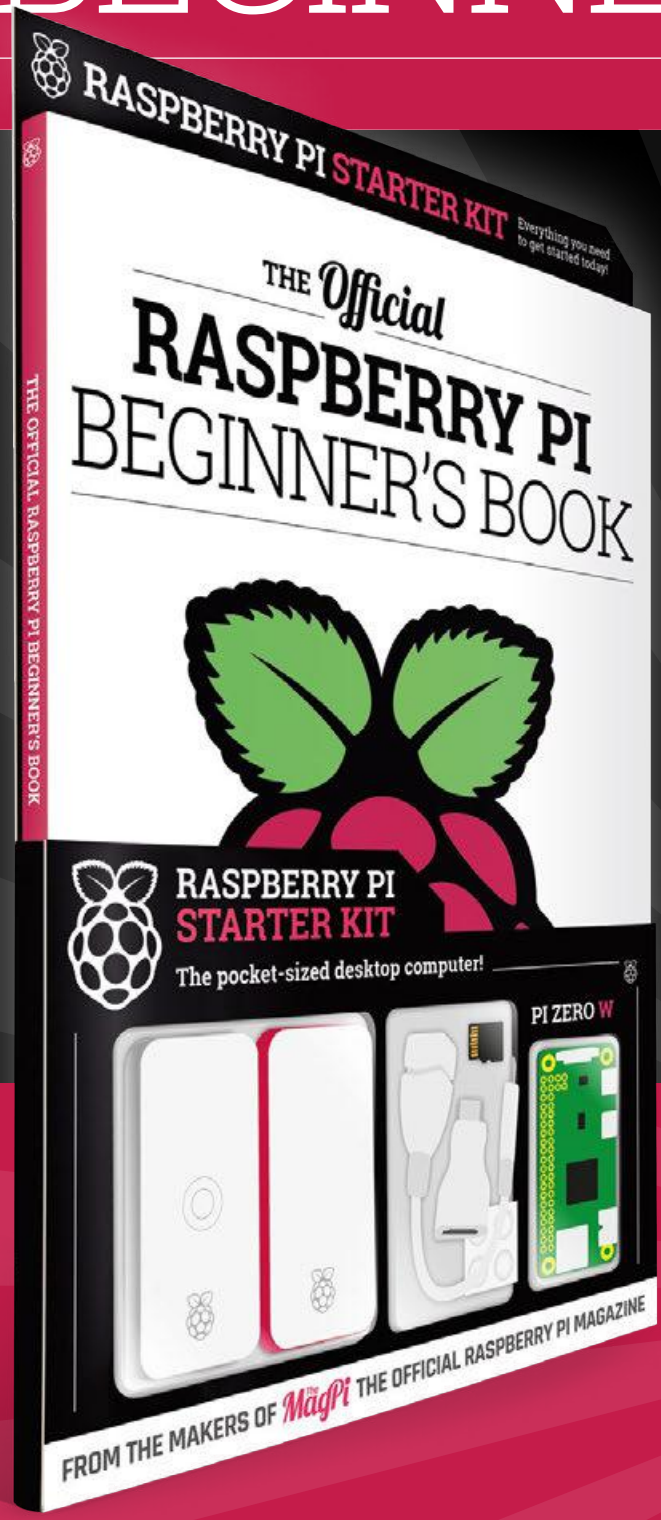
I love projects that display a person's passion for technology and how their interests intersect. We see people using Raspberry Pi for hobbies and disciplines such as beekeeping, automotive hacking, visual arts, beer brewing, and amateur radio. Using Raspberry Pi for making music tends to be very popular. For example, in last month's issue of *The MagPi*, Mike Cook shared how he used Raspberry Pi to create a MIDI drum sequencer. Later in that same issue, there's a profile of musician Yerzmyey, who created an album using Raspberry Pi.

Know the score

I recently came across a Raspberry Pi-based dartboard scoring system made by the user whatarethebands on Reddit (**magpi.cc/rCDjwc**).The Raspberry Pi is rigged up to an electronic dartboard and automatically keeps track of the scores on-screen. I think it's a perfect example of a project that lies at the intersection of two interests. Both the Raspberry Pi community and the darts community on Reddit absolutely loved seeing this project.

If you're anything like me, when you're making at the intersection of Raspberry Pi and your other hobbies, it can motivate you to keep up with both. It also allows for cross-pollination between communities of enthusiasts. What better way to introduce more people to making than when they see what's possible with a Raspberry Pi?

THE *Official* RASPBERRY PI BEGINNER'S BOOK


RASPBERRY PI STARTER KIT
Everything you need to get started today!
THE Official RASPBERRY PI BEGINNER'S BOOK
THE OFFICIAL RASPBERRY PI BEGINNER'S BOOK
RASPBERRY PI STARTER KIT
The pocket-sized desktop computer!
PI ZERO W
FROM THE MAKERS OF THE MagPi THE OFFICIAL RASPBERRY PI MAGAZINE

LEARN COMPUTING THE EASY WAY!

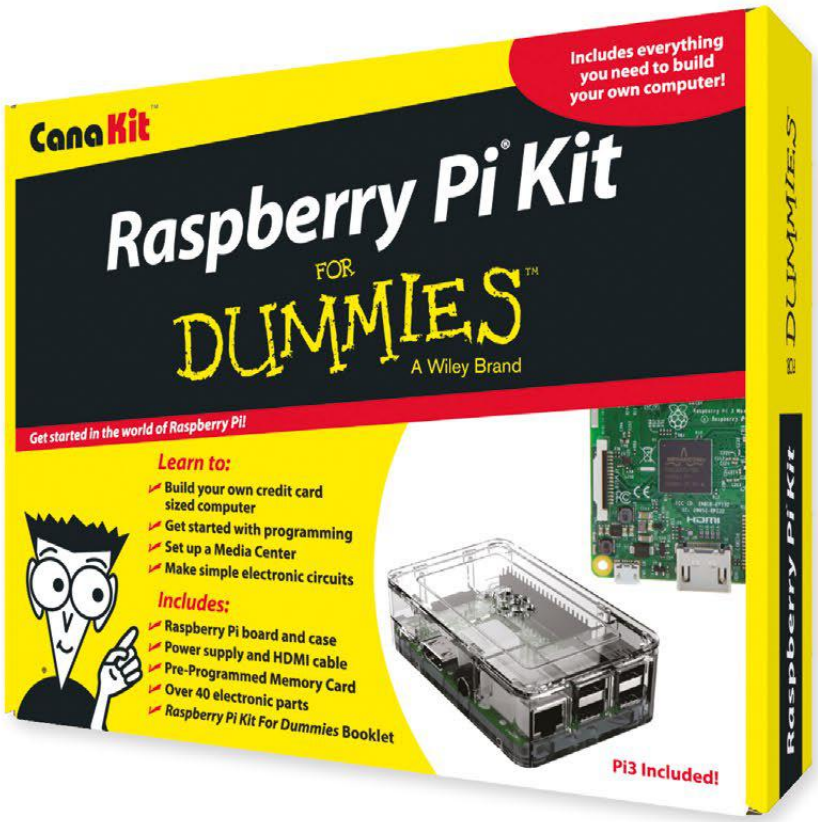
Includes

- Pi Zero W computer
- Official case with three covers
- USB and HDMI adaptors
- 8GB Micro SD card
- 116-page beginner's book

Available now

Buy online: magpi.cc/store

CanaKit™


CanaKit
Includes everything you need to build your own computer!
Raspberry Pi® Kit
FOR DUMMIES™
A Wiley Brand
Get started in the world of Raspberry Pi!
Learn to:
Build your own credit card sized computer
Get started with programming
Set up a Media Center
Make simple electronic circuits
Includes:
Raspberry Pi board and case
Power supply and HDMI cable
Pre-Programmed Memory Card
Over 40 electronic parts
Raspberry Pi Kit For Dummies Booklet
Pi3 Included!
Raspberry Pi Kit

FOR DUMMIES®
A Wiley Brand

Kit Includes:

- *Raspberry Pi For Dummies* Booklet
- Raspberry Pi 3 Board
- Memory Card
- Plastic Case
- 2.5A Power Supply
- HDMI Cable
- Resistors
- LEDs
- Push Button Switches
- Prototyping Breadboard
- Jumper Wires
- Heat Sinks

Available for worldwide shipping at:

WWW.CANAKIT.COM

Available in Europe through RS Components

RS

$89.99 US DOLLARS

£69.99 EXCLUDING VAT

Raspberry Pi is a registered trademark of the Raspberry Pi Foundation. For Dummies and the Dummies Man logo are trademarks or registered trademarks of John Wiley & Sons, Inc. Used under license. RS logo is a registered trademark of RS Components Ltd. CanaKit is a registered trademark of Cana Kit Corporation.

ELECTRONIC KITS • ELECTRONIC PARTS • RASPBERRY PI • ARDUINO